

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Flexible Air Traffic Control Management

Tiago Luís Pacheco Neto



Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Prof. Doutor Daniel Castro Silva

Co-supervisor: Mestre Álvaro Luiz Panarra das Neves Câmara

Julho 18, 1017

Flexible Air Traffic Control Management

Tiago Luís Pacheco Neto

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo Júri:

Presidente: Rosaldo José Fernandes Rossetti

Arguente: Pedro Manuel Henriques da Cunha Abreu

Vogal: Daniel Augusto Gama de Castro Silva

Julho 18, 1017

Resumo

Ano após ano, tem sido observado um crescimento de voos existentes sobre o espaço aéreo europeu, sendo, de acordo com a *Eurocontrol - European Organization for the safety of air navigation*, previsto para 2035 um crescimento na ordem dos 50 pontos percentuais do número total de voos quando comparado com o ano civil de 2012. Ainda segundo a mesma fonte, prevê-se que a capacidade dos aeroportos europeus não irá acompanhar o rápido crescimento do número de voos em condições *IFR - Instrument Flight Rules*. Dado os custos associados ao melhoramento das infraestruturas dos aeroportos é preferível a criação de uma ferramenta rápida, eficaz e estável que seja capaz de auxiliar os controladores de tráfego aéreo a gerirem não só os recursos de um aeroporto como também os voos existentes que possuem a seu cargo de uma forma mais eficiente.

O objetivo nuclear desta dissertação é a diminuição do congestionamento do espaço aéreo, no entanto, tendo a plataforma utilizada o objetivo de simular múltiplas missões de forma realista, foi objetivada a diminuição da discrepância existente entre a realidade e a plataforma. De forma a se conseguir retratar a realidade na plataforma foi necessário a recolha manual de dados relativos a rotas de aproximação e de partida de múltiplos aeroportos, assim como os locais utilizados para a realização de manobras de espera. Foi ainda necessário a introdução de rotas de aproximação/chegada ao espaço aéreo dos aeroportos mapeados e os vários pontos do espaço aéreo norte-americano na região dos grandes lagos. Na tentativa de solucionar o congestionamento do espaço aéreo e o desfasamento da plataforma para com a realidade introduziu-se um novo módulo à componente responsável pela gestão das aeronaves. Este módulo, denominado por *ATC*, apresenta-se como onisciente e encontra-se em contacto permanente com os controladores dos vários aeroportos para que, caso se justifique, seja possível fazer correções e/ou melhoramentos nos voos. Embora parte da solução para o problema seja, de facto, um melhor controlo de tráfego aéreo a nível do *TRACON - Terminal Radar Approach Control*, o mesmo não consegue, por si só, anular o problema. Foi então implementado um módulo responsável pelas manobras de espera, conjunto de manobras usadas para manter o espaço aéreo organizado enquanto as aeronaves aguardam por permissão para aterrar, como forma de se conseguir fazer uma melhor gestão das aeronaves desde o aeroporto de partida até ao de chegada. Como forma de diminuir o número de aeronaves presentes no espaço aéreo monitorizado pelos controladores de aeroporto, para além do uso de rotas de aproximação e de partida reais, foi criado um algoritmo que, prevendo a demanda de recursos, consegue escolher um grupo de pistas a serem usadas de forma a aumentar o fluxo de aeronaves.

Em suma, ao longo desta dissertação conseguiu-se diminuir a discrepância entre a realidade e o existente na plataforma, um melhoramento na taxa de aterragem na ordem dos 25% face ao que já existia, embora se tenham adicionado regras que não existiam, e a criação de um sistema robusto capaz de criar grupos de pistas. Foi ainda criado um controlador capaz de auxiliar e de entregar aeronaves de forma constante os vários aeroportos que estão sobre a sua alçada.

Keywords: Controlo de Tráfego Aéreo, Holding, Espaço Aéreo, Comunicação entre Aeronaves, Congestionamento aéreo

Abstract

Every year, Europe has witnessed a growth in the number of flights in its airspace. According to *Eurocontrol - European Organization for the safety of air navigation*, it is expected a growth of 50 per cent, for the year of 2035, over the total number of flights when compared to the year of 2012. The biggest concern with such growth is the lack of capacity of the airports to manage this air traffic. With that in mind, there is a necessity of creating a fast, stable and efficient platform to help air traffic controllers in their daily activities, such as managing airport resources and/or controlling flights.

The main purpose of this dissertation is the reduction/improvement of the airspace congestion near airports. However, since the Platform was created to simulate multiple missions in a realistic way, it was added the objective of reducing the discrepancy between the reality and the Platform. So it could be possible to reduce the distance between the reality and the Platform, it was necessary to collect several data available on official records. Overall it was collected data about approach and departure routes for several runways of multiple airports, as well as the places used to maintain aircraft on hold. It was also needed to introduce approach / arrival routes to the airspace of the used airports and the various points of the North American airspace in the Great Lakes region.

On the other hand, in order to solve/reduce the airspace congestion it was introduced a new module to the already existing ATC component. As this component presents itself as omniscient and is in permanent contact with airport controllers, it can make slight adjustments in the aircraft's flights. Despite being part of the solution to the congestion problem, the introduction of this module do not solved the problem entirely. It was then implemented a new module responsible for making aircraft able to hold in the airspace while they wait for permission to land. To improve landing rates, it was developed a new way of giving permission to land so several aircraft can be approaching the runway. Finally, it was implemented a new algorithm to create groups of runways that can be used simultaneously so the airport could adapt to its demands.

In short, during this dissertation the discrepancy between the reality and Platform was reduced as an improvement in the landing rate of 25 % compared to what already existed was seen. After the results, it was possible to conclude that the algorithm created to group runways is able to adapt to different runway configurations. It was also created a controller capable of assisting and delivering aircraft steadily to the various airports that are connected to him.

Agradecimentos

Começo por agradecer profundamente ao Professor Doutor Daniel Castro Silva e a Mestre Álvaro Câmara por terem permitido que trabalhasse numa dissertação que tanto me diz e por toda a ajuda e calma que me foram dando ao longo da realização do trabalho.

Gostaria ainda de agradecer aos meus amigos mais próximos João, Rafael, Carolina e Mariana por todos os anos de boas memórias que me proporcionaram e por todo o apoio, força e compreensão que me deram neste último semestre.

Para fim gostaria de agradecer, do fundo do meu coração, aos meus pais por todas os sacrifícios que fizeram ao longo destes 22 anos que permitiram que tivesse uma melhor educação. Agradeço ainda todo o apoio emocional e força que me ofereceram vezes e vezes sem conta e que permitiu o término desta dissertação. Por tudo o que representam na minha vida, é a vós que dedico inteiramente esta dissertação.

Tiago Neto

“I do not think there is any thrill that can go through the human heart like that felt by the inventor as he sees some creation of the brain unfolding to success... such emotions make a man forget food, sleep, friends, love, everything.”

Nikola Tesla

Conteúdo

1	Introdução	1
1.1	Contexto e enquadramento	1
1.2	Motivação	2
1.3	Objectivos	3
1.4	Estrutura do relatório	5
2	Fundamentos	7
2.1	Espaço Aéreo	7
2.1.1	Fases de Voo	7
2.1.2	Classes do Espaço Aéreo	8
2.1.3	ARTCC	10
2.1.4	TRACON	12
2.1.5	ATCT	13
2.1.6	Plano de Voo	14
2.1.7	NOTAM	16
2.1.8	Comunicação ATC-Piloto	17
2.2	Aeroporto	20
2.2.1	Pista	20
2.2.2	Taxiway	23
2.2.3	Cartas Aéreas de Aproximação	24
2.3	Manobra de Espera standard	26
2.3.1	Entradas para a Manobra de Espera	27
2.3.2	Regras para Manobra de Espera pela <i>FAA</i>	28
2.4	Diferenças entre <i>FAA</i> e as recomendações da <i>ICAO</i>	31
2.4.1	Comunicação	31
2.4.2	Combustível	31
2.4.3	Manobra de Espera	32
3	Revisão da Literatura	33
3.1	Congestão do Espaço Aéreo	33
3.1.1	Planeamento de voo	33
3.1.2	Otimização de Aterragens	35
3.1.3	Otimização e criação de rotas aéreas	37
3.2	Otimização do espaço aéreo	45
3.3	Plataformas em uso	47
3.3.1	NAVCANatm	47
3.3.2	STARS (<i>Standard Terminal Automation Replacement System</i>)	48
3.3.3	VATSIM	48

CONTEÚDO

3.3.4	NextGen	49
3.3.5	SESAR	50
4	Planeamento	53
4.1	Tecnologias Utilizadas	53
4.1.1	Microsoft Flight Simulator	53
4.1.2	AgentService	54
4.2	Abordagem ao problema	54
4.2.1	Diminuição do congestionamento do tráfego aéreo	54
4.2.2	Lidar com o congestionamento aéreo	55
4.3	Planeamento e Análise de risco	56
5	Implementação	59
5.1	Arquitetura do sistema	59
5.1.1	Agente ATC	60
5.2	Rotas de aterragem e de partida das Aeronaves	62
5.3	Manobras de Espera	65
5.3.1	Escolha de pistas a serem utilizadas	65
5.3.2	Gestão das aeronaves em espera	69
5.4	Criação de um novo controlador	74
5.5	Alteração das Linguagens Utilizadas	80
5.5.1	<i>Scenario</i>	82
5.5.2	<i>Mission</i>	86
5.6	Alterações visuais nos vários componentes	87
6	Resultados	91
6.1	Construção de Grupos de Pista	91
6.1.1	Aeroportos Utilizados	91
6.1.2	Resultados	93
6.2	Gestão de Filas de Aterragem em Caso de Congestionamento	95
6.2.1	Cenários de Testes	95
6.2.2	Resultados	95
6.3	Gestão do Espaço Aéreo do Aeroporto com Auxílio do <i>TRACON</i>	97
6.3.1	Cenários de Teste	98
6.3.2	Resultados	98
7	Conclusões e Trabalho Futuro	101
7.1	Conclusões	101
7.2	Trabalho Futuro	102
	Referências	103
A	Anexos	109
A.1	Alterações nos Ficheiros de Linguagem	109
A.1.1	Ficheiros relativos à <i>SDL</i>	109
A.1.2	Ficheiros relativos à <i>MDL</i>	113

Lista de Figuras

1.1	Movimentos <i>IFR</i> totais na RIV de Lisboa	2
2.1	As várias fases de voo existentes.	8
2.2	Figura ilustrativa das classes existentes nos espaços aéreos.	9
2.3	Demonstração da Divisão do Espaço Aéreo Americano	11
2.4	Representação das diferenças dos vários sectores existentes.	13
2.5	Exemplo de um plano de voo.	14
2.6	Exemplo de uma pista com as respetivas marcações.	20
2.7	Exemplo de uma <i>single runway</i>	21
2.8	Demonstração de possíveis configurações de pistas	22
2.9	Exemplo de uma configuração <i>open-v</i>	22
2.10	Exemplo de uma <i>Intersecting runways</i>	23
2.11	Figura ilustrativa de várias <i>taxiway</i>	24
2.12	Carta de aproximação à pista 6L do aeroporto <i>KCLE</i> , Ohio.	25
2.13	Carta de aproximação ao aeroporto <i>KCLE</i> , Ohio.	26
2.14	Descrição dos pontos referentes à manobra de <i>Holding</i>	27
2.15	Figura ilustrativa de como entrar na manobra de <i>Holding</i>	28
2.16	Figura ilustrativa dos vários entradas usadas para iniciar a manobra de espera. . .	29
2.17	Figura ilustrativa de um Local para a Realização de Manobras de Espera.	29
3.1	Teste efetuado pelos autores do artigo.	34
3.2	Exemplo de um planeamento viável usando 6 aeronaves e 2 <i>janelas</i>	36
3.3	Exemplo dado pelos autores de um <i>grid</i> gerado pelo algoritmo em fases iniciais. .	39
3.4	Exemplo de uma extensão de nó dada pelos autores.	40
3.5	Exemplo de um <i>Bucket-shaped area</i> dada pelos autores.	41
3.7	Exemplo dado pelos autores de uma <i>grid</i> com estados (x) e estágios (J).	43
3.8	Diferença entre distâncias das rotas geradas pelo algoritmo e as rotas pré-definidas pela <i>FAA</i>	44
3.9	Exemplo dado pelos autores de uma área onde são estudadas as rotas.	45
3.10	Exemplo dado pelos autores de condições previstas Vs condições reais.	46
3.11	Exemplo dado pelos autores de condições previstas Vs condições reais.	47
3.12	Arquitetura do modelo criado pelos autores do estudo.	48
3.13	Criação de nós ao longo das regiões norte-americanas em exemplo dado pelos autores.	49
3.14	Desvio na Rota calculado através do problema linear em exemplo dado pelos autores.	49
3.15	Conjunto de gráficos representativos dos resultados obtidos.	50
3.16	Figura ilustrativa da informação demonstrada pelo <i>STAR</i>	51
3.17	Exemplificação de uma simulação no <i>VATSIM</i>	52

LISTA DE FIGURAS

4.1	Exemplo da estratificação usada na gestão de aeronaves em espera.	55
4.2	Planeamento das tarefas a realizar dispostas num gráfico Gantt	56
5.1	Arquitetura de componentes que compõem a plataforma.	59
5.2	Arquitetura dos módulos que constituem a componente do ATC.	61
5.3	Rota de aproximação, base para criação do polígono, da pista 6L de Cleaveland, OHIO à esquerda, e polígono consequente à direita.	66
5.4	Polígono resultante para a pista 6L de Cleveland, Ohio para a operação de partida.	67
5.5	Exemplo de uma ligação no grafo mostrando que os as pistas 6L <i>Approach</i> e 6R <i>Approach</i> podem ser usadas em simultâneo.	69
5.6	Exemplo de um caso possível que levaria à criação de dois grupos.	70
5.7	Exemplo da disposição dos pontos entre uma manobra de espera <i>standard</i> e <i>non standard</i>	70
5.8	Exemplo da lógica após saída de uma aeronave do local de espera.	71
5.9	Máquina de estados de uma aeronave já com os novos estados adicionados.	72
5.10	Exemplo de uma possível troca de mensagens entre agentes veículos e um agente ATC.	73
5.11	Exemplo da aérea controlada pelo controlador do tipo <i>Tracon</i> mapeado.	75
5.12	Exemplo de mensagens trocadas num cenário em que a aeronave transita do espaço aéreo do controlador <i>ATC 1</i> para o <i>ATC 2</i>	76
5.13	Rota de um dos <i>STARS</i> com manobras de espera conectadas entre si.	77
5.14	Máquina de estados para as aeronaves geridas pelo novo controlador.	80
5.15	Representação gráfica dos atributos que constituem a secção <i>Airspace</i> na <i>SDL</i>	82
5.16	Representação gráfica do atributo <i>waypoints</i> na secção <i>Airspace</i>	83
5.17	Representação gráfica do atributo <i>airway</i> na secção <i>Airspace</i>	83
5.18	Descrição visual dos elementos existentes em <i>approachRoutes</i>	84
5.19	Representação gráfica dos atributos existentes em <i>approachWaypoint</i>	84
5.20	Representação da componente de <i>departures</i>	85
5.21	Representação das várias possibilidades de uma <i>sequenceAlternative</i>	85
5.22	Representação da componente <i>star</i>	85
5.23	Representação da componente <i>courses</i>	86
5.24	Representação da componente <i>courses</i>	87
5.25	Representação do espaço aéreo para um controlador do tipo <i>Tracon</i>	88
5.26	Exemplo de informações que podem estar representadas nas várias tabelas existentes.	89
5.27	Novo separador contendo a funcionalidade de controlo de múltiplas aeronaves de forma simultânea.	89
6.1	Configuração do aeroporto <i>KCMH</i>	92
6.2	Configuração do aeroporto <i>KCAK</i>	92
6.3	Configuração do aeroporto <i>KCLE</i>	93

Lista de Tabelas

2.1	Descrições das Zonas Especiais do Espaço Aéreo Norte Americano	11
2.2	Velocidade Máxima permitida por altitude e tipo das aeronaves	30
2.3	Limite máximo de velocidades por altura	32
4.1	Tabela dos vários riscos presentes no desenvolvimento da presente dissertação. .	57
5.1	Exemplo de aproximações às pistas 6L e 6R do aeroporto de Cleveland, OHIO. .	62
5.2	Exemplo de rotas de partida no aeroporto de Cleveland, OHIO.	64
5.3	Exemplo dos vários pontos perto do aeroporto de Cleveland, OHIO onde se é capaz de realizar manobras de espera.	77
6.1	Resultados obtidos durante a criação de grupos	93
6.2	Resultado dos valores dados aos vários grupos de pistas para cada cenário. . . .	94
6.3	Tabela demonstrativa das pistas presentes nos melhores e piores grupos para cada tuplo Aeroporto-Cenário.	94
6.4	Resultados obtidos relativos às taxas de aterragem relativas ao aeroporto <i>KCMH</i> . .	96
6.5	Tempo que as últimas aeronaves esperam entre pedir para aterrar e obterem permissão no aeroporto de <i>KCMH</i>	96
6.6	Resultados obtidos relativos às taxas de aterragem relativas ao aeroporto <i>KCLE</i> . .	96
6.7	Tempo que as últimas aeronaves esperam entre pedir para aterrar e obterem permissão no aeroporto de <i>KCLE</i>	97
6.8	Resultados obtidos relativos às taxas de aterragem relativas ao aeroporto <i>KCAK</i> . .	97
6.9	Tempo que as últimas aeronaves esperam entre pedir para aterrar e obterem permissão no aeroporto de <i>KCAK</i>	97
6.10	Resultados obtidos no aeroporto de destino com auxilio do <i>TRACON</i>	98

LISTA DE TABELAS

Lista de Listagens

5.1	Exemplo das <i>tags</i> resultantes a adicionar ao aeroporto de Cleveland, OHIO. . . .	63
5.2	Exemplo de rotas de partida no aeroporto de Cleveland, OHIO em formato XML.	63
5.3	Exemplo da estruturação dos dados relativos ao espaço aéreo	76
5.4	Exemplo da estruturação dos dados relativos aos <i>stars</i> nas proximidades do aeroporto de Cleveland, Ohio	78
5.5	Exemplo da estruturação de uma rota	78
A.1	Alterações no ficheiro <i>scenario.xsd</i>	109
A.2	Alterações no ficheiro <i>mission.xsd</i>	113

LISTA DE LISTAGENS

Acrónimos

API	Application programming interface
ARTCC	Air Route Traffic Control Center
ATCT	Air Traffic Control Tower
ATRTLP	Average Time from Request To Land Permission
CIWS	Corridor Integrated Weather System
CSV	Comma-Separated Values
CWAM	Convective Weather Avoidance Model
DME	Distance Measure Equipment
FAA	Federal Aviation Administration
FCFS	First Come First Served
FIFO	First In First Out
FIPA	Foundation for Intelligent and Physical Agent
FSX	Flight Simulator X
FT	Feet
GMT	Greenwich Mean Level
IATA	International Air Transport Association
ICAO	International Civil Aviation Organization
IFR	Instrument Flight Rules
KCAK	Akron-Canton Regional Airport
KCLE	Cleveland-Hopkins International Airport
KCMH	John Glenn Columbus International Airport
KJFK	John F. Kennedy International Airport
LIACC	Laboratório de Inteligência Artificial e Ciência de Computadores
MDL	Mission Description Language
MOA	Military Operations Area
MSL	Mean sea level
NAS	National Airspace System
NASA	National Aeronautics and Space Administration
NextGen	Next Generation Air Transportation System
RIV	Regiões de Informação de Voo
RNAV	Area Navigation
SAS	Sparse A* Search
SAW	Simple Additive Weighting
SDL	Scenario Description Language
SESAR	Single European Sky ATM Research
STAR	Standard Terminal Automation Replacement System
TRACON	Terminal Radar Approach Control Facilities
USAF	United States Air Force

Acrónimos

VATSIM	Virtual Air Traffic Simulation Network
VFR	Visual Flight Rules
VoIP	Voice over IP
XML	eXtensible Markup Language
XSD	XML Schema Definition

Capítulo 1

Introdução

Desde 1905, início da era moderna da aviação começada com o voo dos irmãos Wright [Shaw, 2014], foram sendo aprimoradas e criadas cada vez mais aeronaves capazes de transportar carga e/ou passageiros de uma forma eficiente e rápida, tendo então sido criada uma nova forma de transporte. Com o rápido crescimento que se verificou e que ainda é visível foi fundamental a criação de métodos capazes de orientar e de salvaguardar a segurança das aeronaves. Serve o presente capítulo para demonstrar os problemas, as necessidades e quais os objetivos que se pretende resolver/ implementar nesta dissertação.

1.1 Contexto e enquadramento

A comodidade de poder fazer a ligação de diferentes pontos do mundo, muitas vezes distanciados por milhares de quilómetros, num curto período de tempo, permitiu que o meio de transporte aéreo se fosse tornando cada vez mais indispensável quer num foro comercial, quer num foro pessoal e de lazer, de tal forma que, de acordo com [Morris, 2017], aproximadamente três mil e setecentos milhões de passageiros cruzaram os céus. O crescimento do meio de transporte aéreo tem sido uma constante na época atual que embora não seja exponencial tem sido linear. Usando os dados disponibilizadas pela NAV Portugal, entidade responsável pela prestação de serviços de tráfego aéreo em território português, é possível observar a existência de um crescimento na casa dos 10.6 pontos percentuais do número total de voos em circunstâncias *IFR* (*Instrumental Flight Rules - Regras de Voo por Instrumentos*), sendo que esta estatística apenas é representativa da realidade vivida no ano transato em Portugal Continental e Madeira. Tal é comprovado através da Fig. 1.1 [NAV Portugal, 2017]. Como está demonstrado na mais recente previsão divulgada pela *Eurocontrol*, o crescimento não se tem verificado apenas no país lusitano, mas em quase toda a Europa, sendo ainda esperado, que ano após ano, sejam batidos recordes[Eurocontrol, 2016].

Em conjunto com a comodidade temporal sob longas trajetórias fornecida pelo meio de transporte aéreo, existe a segurança que este tipo de viagens oferece. Segundo a *CNN*, canal de informação norte-americano, o transporte aéreo é o mais seguro de todos os tipos de transporte com 0.06 mortes por cada mil milhões de quilómetros [Isidore, 2015]. Um dos grandes pilares em que

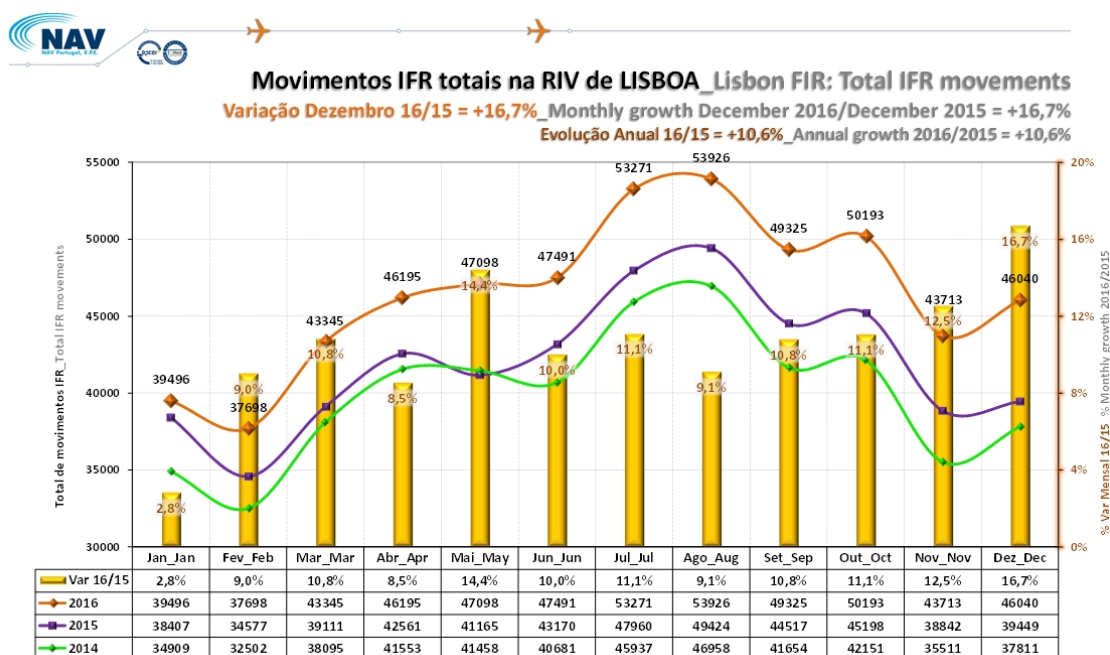


Figura 1.1: Movimentos *IFR* totais na RIV (Região de Informação de Voo) de Lisboa, retirado de [NAV Portugal, 2017]

assenta a segurança dos transportes aéreos são os controladores que fornecem durante 24 horas, um serviço capaz de auxiliar as aeronaves em todos os períodos das suas viagens, começando no aeroporto de origem e terminando no aeroporto de destino. Ao aliar o crescente adensamento do tráfego aéreo ao já existente, que se traduz num maior número de aviões à sua responsabilidade, aos inevitáveis problemas de foro ambiental, como o vulcão Eyjafjallajökull que em Abril de 2010 levou ao fecho da maioria dos espaços aéreos europeus, os controladores de tráfego aéreo vêm a sua tarefa de manter a segurança de todos os presentes no seu espaço aéreo ainda mais complexa. Torna-se, assim, totalmente necessário melhorar e/ou criar ferramentas capazes de facilitar o cumprimento das tarefas dos controladores. Este projeto encontra-se inserido numa plataforma, cujo objetivo passa pela capacidade de simulação realista de diversos tipos de missões, que tem vindo a ser desenvolvido no LIACC (Laboratório de Inteligência Artificial e Ciências de Computadores) e já serviu como base para múltiplas dissertações como [Camara, 2013], [Silva, 2011], [Santos, 2010], [Silva, 2008] e [Sousa, 2010]. Para a presente dissertação o objetivo máximo passa pela prestação de auxílio aos controladores de tráfego aéreo nas suas tarefas diárias sendo possível aos mesmos estarem atentos e prestar auxílio imediato a situações que realmente careçam de atenção.

1.2 Motivação

Como já foi salientado e demonstrado na secção anterior, a necessidade e responsabilidade que recai sobre os transportes aéreos tem vindo a sentir um crescimento constante e gradual. É

nesta premissa fundamental que se concentra a motivação base para a realização desta dissertação. Possuindo as quatro dimensões, três dimensões espaciais em conjunto com a dimensão temporal, uma enorme relevância na atualidade da aviação moderna, tornou-se fulcral a capacidade de gerir eficazmente não só o espaço em torno dos aeroportos em manobras como a de espera (*holding* na nomenclatura inglesa), como antecipar possíveis contratempos e automatizar processos capazes de facilitar o trabalho dos controladores aéreos.

Assim e com o objetivo de traduzir as necessidades de todos os envolvidos nos transportes aéreos, desde os passageiros, controladores aéreos até aos pilotos, foi ambicionado uma plataforma capaz de gerir de forma correta e eficaz as aeronaves em diferentes estados do seu voo.

1.3 Objectivos

Esta dissertação tem como meta a automatização de algumas das tarefas que são praticadas regularmente pelos controladores aéreos, de forma a possibilitar uma melhor gestão do espaço aéreo. O presente trabalho dá continuidade ao que foi sendo feito na plataforma, em que tarefas como a gestão de aeronaves quer em terra quer no ar, deteção de possíveis colisões entre aeronaves, e a possibilidade de configurar diferentes aeroportos foram desenvolvidas. Com o intuito de aproximar o projeto de uma plataforma estável e capaz de gerir aeronaves desde o momento em que a sua viagem tem início até ao seu término, pretende-se implementar as seguintes funcionalidades:

- **Diminuição do congestionamento do tráfego aéreo**

O objetivo basilar desta dissertação é a diminuição do congestionamento nas proximidades dos aeroportos. Dada a grandeza do objetivo, subdividiu-se o mesmo em tarefas mais específicas.

- **Comunicação entre aeroportos**

Atualmente, o cumprimento do horário de chegada é, entre outras como suporte ao passageiros e números de destinos possíveis, umas das métricas utilizadas para a avaliação de companhias aéreas por parte dos clientes. Ocasionalmente, as condições mínimas para a aterragem das aeronaves são inexistentes e, como tal, é emitida a informação que as aeronaves devem esperar, praticando por exemplo a manobra de *holding*, que se encontra descrita no capítulo 2.1, ou tentarem a aterragem noutro aeroporto de destino. Para além de causar transtorno aos passageiros, também o causa às companhias, que para além de desperdiçarem combustível, atrasam o início e consequente fim das próximas viagens.

O que se pretende implementar neste módulo é a comunicação entre aeroportos, de forma a que seja possível às torres de controlo, caso seja expectável um congestionamento no seu espaço aéreo, atrasar a data de partida dos voos. Com esta funcionalidade não só seria possível poupar combustível às companhias aéreas que, de outra forma, seria desperdiçado como também minimizar a quantidade de tráfego aéreo existente no

espaço aéreo de um aeroporto, facilitando assim o trabalho exigente dos controlares aéreos.

– **Melhoria nas manobras de espera e melhoria na eficiência das pistas de aterragem**

Com a contribuição de [Sousa, 2010], a funcionalidade de emitir uma ordem de realização de uma manobra de espera e a sua realização por parte da aeronave já está implementada. Contudo, ao contrário do que acontece na realidade, apenas existe um nível para a realização da manobra de espera, não sendo então possível diferentes aeronaves realizarem as manobras no mesmo ponto bidimensional com diferentes altitudes, o que causa, com o prolongar do tempo, uma saturação do espaço aéreo. Por vezes é necessário conceder permissão para aterrar a aeronaves que em considerações normais teriam de esperar que outras aeronaves aterrassem. Exemplo dos vários fatores que implicam uma alteração na ordem de aterragem são, por exemplo, a existência de uma menor quantidade de combustível ou a existência de uma emergência médica a bordo. Será uma das metas desta dissertação, como forma de conseguir descongestionar o espaço aéreo, a implementação de manobras de espera estratificadas em altura sendo que aeronaves que possuam prioridades/emergências obtenham permissão o mais rápido possível. É também desejado uma melhoria na performance da pista de aterragem através de uma melhor sequencialização das aeronaves a aterrar assim como uma escolha apropriada de pistas a serem utilizadas.

– **Alteração de rotas**

Por vezes, devido a situações imprevistas, de causa ambiental ou humana, ou apenas com o intuito de facilitar o trabalho dos controladores aéreos e/ou dos pilotos, é necessária a execução de uma alteração de rota. Esta funcionalidade pode ser usada, por exemplo, no caso do avião já se encontrar em viagem para o aeroporto de destino, aquando da ocorrência de algo que impeça as aeronaves de aterrarem no período desejado. Neste caso, de forma a evitar um congestionamento do espaço aéreo do aeroporto, é alterada a rota às aeronaves de formas a que estas atinjam o seu destino num horário em que a sua aterragem será facilitada. Um problema diferente é, com uma solução semelhante, a necessidade de alterar a rota de uma aeronave em caso de conflito com outra aeronave, sendo depois necessário o retorno à rota inicial após aliviado o conflito. É então neste contexto, que se pretende implementar um módulo capaz de alterar a rota de aeronaves com o intuito de atrasar ou de adiantar a sua chegada ao aeroporto de destino.

● **Alteração de regras entre ICAO e FAA**

A maioria das recomendações dadas pela organização *ICAO - International Civil Aviation Organization* já foram aceites e encontram-se em uso na maioria das nações. Contudo países como Estados Unidos da América possuem pequenas diferenças face às recomendações propostas pela *ICAO*. Visto que no estado atual do projeto estão em uso regras usadas pela

FAA - Federal Aviation Administration e dada a importância crescente das recomendações usadas pela organização internacional, é pretendida a implementação de um conjunto de regras recomendadas por esta última de forma a que o projeto consiga simular qualquer aeroporto e usando qualquer tipo de regras.

1.4 Estrutura do relatório

A presente dissertação está subdividida em sete capítulos, começando pelo capítulo atual onde é descrito o contexto, motivação e objetivos. No Capítulo 2 é feita uma introdução ao mundo da aviação, sendo que se começa por descrever os vários centros responsáveis pela gestão, em diferentes momentos da viagem, das aeronaves. Para além de ser feita uma distinção entre as regras mais comuns nos aeroportos internacionais, é ainda feita uma descrição de algumas das infra-estruturas existentes nos aeroportos assim como uma descrição da manobra à qual as aeronaves recorrem para se manterem no ar enquanto esperam por permissão para aterrar. No capítulo 3 é descrito o que de melhor foi encontrado sobre os vários problemas a resolver. Para este capítulo foram estudados *softwares* já existentes assim como abordagens aos problemas semelhantes ou iguais aos quais se pretende solucionar. No Capítulo 4 é feita uma breve descrição às várias tecnologias que serão usadas no decorrer desta dissertação. É ainda neste capítulo que é referida a abordagem a seguir para que se solucione o problema maior deste trabalho. Já no capítulo 5 é explicado ao leitor o que foi implementado na plataforma no decorrer da presente dissertação, sendo que para além das novas funcionalidades é também explicado os vários procedimentos como recolha de dados e conversão de dados para formatos já em uso. No Capítulo 6, serão discutidos os vários resultados que se obtiveram com o que foi sendo implementado e descrito no Capítulo 5. Por fim e já no Capítulo 7 da presente dissertação conclui-se o trabalho tirando conclusões assim como se sugere futuras implementações capazes de trazer valor à plataforma.

Introdução

Capítulo 2

Fundamentos

De acordo com a mais recente previsão publicada pela *Eurocontrol* é estimado um crescimento de 2.1 pontos percentuais por ano no número total de voos até 2022 para países constituintes da União Europeia. Segundo o mesmo estudo, é esperado um aumento entre 250 a 500 [Eurocontrol, 2016, fig 25] voos adicionais para Portugal, o que será um aumento considerável de tráfego no espaço aéreo dos aeroportos portugueses.[Eurocontrol, 2016] Com o aumento de tráfego aéreo previsto, a melhoria dos vários procedimentos de controlo aéreo torna-se indispensável, de forma a garantir a segurança das aeronaves e dos seus ocupantes.

2.1 Espaço Aéreo

Um espaço aéreo é uma porção tridimensional da atmosfera localizada sobre um território, marítimo e/ou terrestre, que é controlado pelo país detentor desses mesmo territórios. Citando a organização americana, *Federation Aviation Administration (FAA)*, por dia, existem cerca de 23000 voos comerciais, sendo estimado que cerca de 7000 aeronaves estão no céu a qualquer momento do dia. Ao aliar com a grandeza do espaço aéreo americano, 5 milhões de quilómetro quadrados, torna-se impossível uma gestão centralizada de todas as aeronaves [FAA, 2016a]. De forma a colmatar o problema, o espaço aéreo norte-americano foi dividido por 21 zonas [Vanhoenacker, 2015], controladas por ARTCC - (*Air Route Traffic Control Center*), em que cada uma destas zonas são subdivididas em 12 a 25 sectores que controlam regiões perto de aeroportos [Mogford et al., 1995].

2.1.1 Fases de Voo

Desde do seu aeroporto de origem até ao seu destino, uma aeronave atravessa por várias fases de voo. Embora exista um conjunto alargado de fases (treze de acordo com [Skybrary, 2010]), as mesmas podem ser condensadas para somente seis: *TakeOff*, *Departure*, *En-Route*, *Arrival*, *Approach* e *Landing* (ver Fig.2.1).

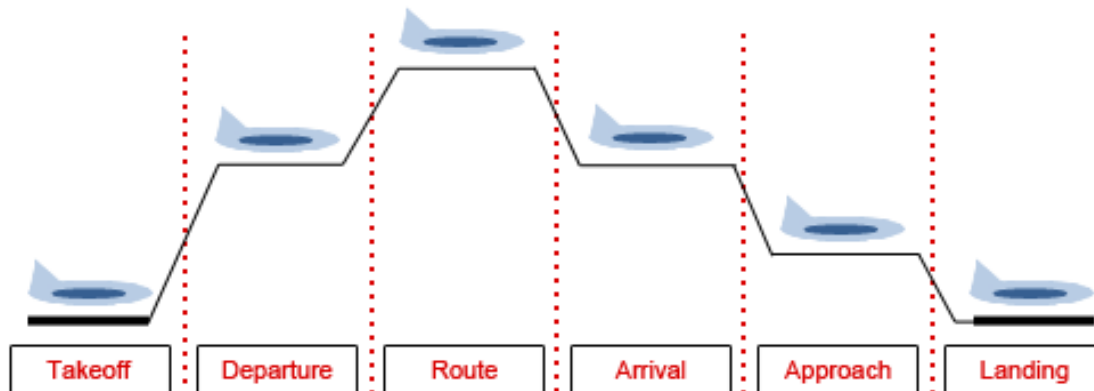


Figura 2.1: As várias fases de voo existentes de acordo com [ENAIRE, 2017].

1. *TakeOff*

Fase que abrange todas as movimentações de uma aeronave desde que lhe é dada permissão para se deslocar para uma pista até que atinge os quatrocentos pés de altitude.

2. *Departure*

Uma aeronave encontra-se na presente fase após ter atingido os quatrocentos pés de altitude após levantar até atingir o primeiro ponto da sua rota.

3. *En-Route*

Nesta fase as aeronaves vão seguindo a sua rota até às proximidades do aeroporto de destino.

4. *Arrival*

Fase na qual as aeronaves começam a diminuir a sua altitude para que se possam aproximar do seu aeroporto de destino.

5. *Approach*

Após ter cruzado o último ponto da sua rota, que inclui as rotas de aproximação ao aeroporto, a aeronave encontra-se na fase de *Approach* até que consiga aterrar.

6. *Landing*

Uma aeronave encontra-se em fase de *Landing* após a sua aterragem até chegar ao hangar.

2.1.2 Classes do Espaço Aéreo

Em 1990, a *International Civil Aviation Organization (ICAO)* adotou um novo sistema de divisão do espaço aéreo. Com o objetivo de melhorar a organização e uniformizar com o resto do mundo o seu espaço aéreo que continha até à data 20 classificações diferentes, a *FAA* adotou o sistema em uso pela *ICAO* em 1993. Com a adoção do sistema criado pela organização europeia

ICAO, o espaço aéreo norte americano foi categorizado em sete secções, sendo que como base para estas diferenciações teve-se em conta, o tipo de navegação - *Instrument Flight Rules (IFR)* ou *Visual Flight Rules (VFR)*, a que altitudes as aeronaves têm permissão para circular e que tipo de controlo existe nesse espaço - se o *Air Traffic Control (ATC)* disponibiliza os seus serviços ou não há uma entidade que controla o espaço aéreo [Nolan, 2010]. Pode, qualquer espaço aéreo (ver Fig. 2.2), ser então classificado em:

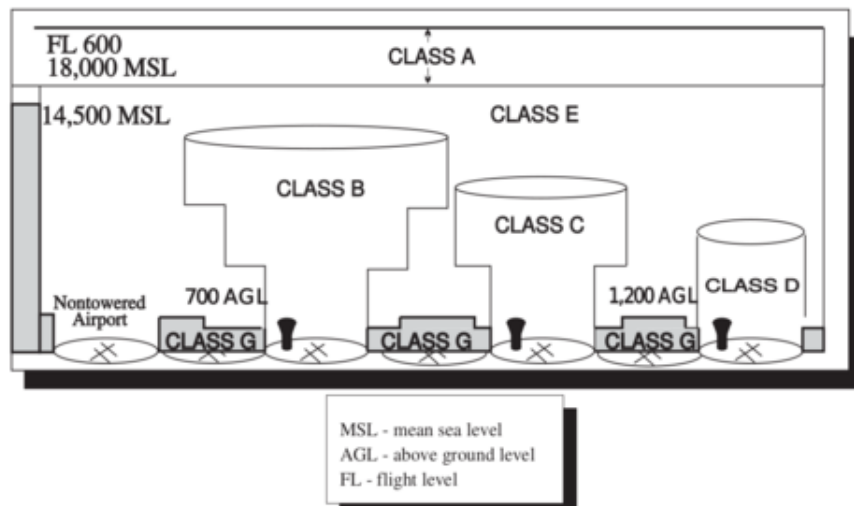


Figura 2.2: Figura ilustrativa das classes existentes nos espaços aéreos conforme em [FAA, 2015a].

- **Controlled Airspace**

Este espaço aéreo representa quase a totalidade do espaço que é navegável pelas aeronaves quer as mesmas sigam em circunstâncias *IFR* ou *VFR*. De acordo com a *FAA*, ser controlado significa que comunicações via rádio, ajudas na navegação e a separação entre aeronaves é suportada e possível servir às aeronaves que naveguem sob *IFR*. É nesta secção, onde se encontram a maior parte das classes criadas em 1990 pela *ICAO*. No total, existem 5 classes pertencentes a este tipo de espaço aéreo - classe A, B, C, D, E.

- **Classe A**

A primeira classe estende-se desde os 18000ft até aos 60000ft *MSL - Medium Sea Level*. Apenas aeronaves que naveguem sob *IFR* são permitidas e, é de salientar que esta classe abrange a fase de voo *en route* e que cabe ao *ATC* auxiliar as aeronaves a separarem-se umas das outras.

- **Classe B**

Nesta classe todas as aeronaves são permitidas quer estejam sob *IFR* ou *VFR*, sendo a única exigência a obrigatoriedade de possuir equipamentos para comunicações via rádio e de navegação. Este classe do espaço aéreo estende-se até aos 12000ft de altitude

e possui um raio até 55km de distância de um aeroporto. Nesta área, o *ATC* pratica *Positive control*, isto é, está responsável pela separação das aeronaves.

– *Classe C*

Zona do espaço aéreo que é definida, à semelhança da classe B, nas proximidades de um aeroporto, embora esta área termine nos 4000ft de altura. Dentro desta área, o *ATC* auxilia as aeronaves sob *IFR* a separarem-se das restantes, *IFR* ou *VFR*, no entanto, aeronaves *VFR* estão dependentes de si manterem separadas de outras aeronaves *VFR*.

– *Classe D*

A classe D, ao contrário das anteriores, não é controlada pelo *ATC*, mas sim, pelas torres de controlo existentes nos aeroportos. A existência desta classe deve-se então à necessidade de uma maior facilidade de controlo das aeronaves nas imediações dos aeroportos. Uma aeronave deve, assim que entre no espaço aéreo, manter contacto com a torre de controlo. Este espaço aéreo tem um raio de 8km e chega a 2500ft de altura.

– *Classe E*

A classe E é a última classe que representa espaço aéreo controlado. Embora o *ATC* forneça a separação de aeronaves, as que se encontram sob *VFR* não são obrigadas a usar o serviço. Os espaços aéreos classificados com esta classe não são delimitados horizontalmente, ao contrário dos classificados como B ou D, mas sim verticalmente, podendo estender-se desde os 700-1200ft até aos 18000ft.

● *Uncontrolled Airspace*

Ao contrário do espaço aéreo controlado, a *FAA* não providencia qualquer serviço às aeronaves, sendo que as mesmas estão responsáveis pela separação entre aeronaves e pela sua navegação. De forma a categorizar estes espaços aéreos utiliza-se a classe F e G, sendo que ambas se distinguem pelo tipo de separação entre aeronaves prestado pelos *ATC*. Enquanto que a primeira classe, F, os controladores podem, se possível, garantir a separação de aeronaves, na segunda não existe qualquer possibilidade de ajuda por parte dos *ATC*.

● *Special Use Airspace*

Existem zonas do espaço aéreo norte-americano que foram designadas para um uso especial, sendo que as mesmas podem ser divididas em diferentes tipos, como *Prohibited*, *Restricted*, *Warning*, *Alert*, *Military Operations Area*. Na Tabela 2.1 são mostradas as diferenças entre os vários tipos.

2.1.3 ARTCC

Os *Air Route Traffic Control Center (ARTCC)*, frequentemente conhecidos por *centers*, têm como principal objetivo providenciar auxílio e orientação a qualquer aeronave que sobrevoe o seu espaço aéreo, seja ela de origem militar, privada ou comercial. Estes centros, onde se controlam

Tabela 2.1: Descrições das Zonas Especiais do Espaço Aéreo Norte Americano

	Descrição
Prohibited	Zonas onde as aeronaves, independentemente do seu tipo, não podem circular. Exemplos destas zonas, é por exemplo, o espaço aéreo acima da Casa Branca.
Restricted	Zonas que, embora não sejam proibidas, podem ter certas restrições devido à possibilidade da existência de riscos.
Warning	Zonas em espaço aéreo internacional, onde a é recomendado a não utilização do espaço devido aos riscos que implicam.
Alert	Zonas que são marcadas nos documentos de aviação para que os pilotos saibam da possibilidade de existirem actividades não usuais no espaço aéreo como, por exemplo, saltos de paraquedas.
Military Operations Areas	Zonas onde treinos e outras manobras militares ocorrem. Aeronaves sob VFR não são impedidas de viajar neste espaço, embora sejam aconselhadas a não o fazer.

aeronaves em fase *en route*, são responsáveis pela gestão do tráfego aéreo sobre uma região do espaço aéreo americano. Na Fig. 2.3 pode ver-se o espaço aéreo americano com cada zona delimitada, sendo que todos os centros são identificados por 3 letras em que a primeira é 'Z'. Estes centros, ao contrário dos *TRACONs* - (*Terminal Radar Approach Control*) como veremos posteriormente, estão sempre operacionais, independentemente da hora ou dia. Os controladores de tráfego aéreo que operam nestes centros monitorizam as aeronaves através de radares e, caso sejam precisos ajustes ao itinerário, comunicam via rádio através de frequências previamente acordadas com os pilotos das aeronaves de modo a manter a segurança da mesma.

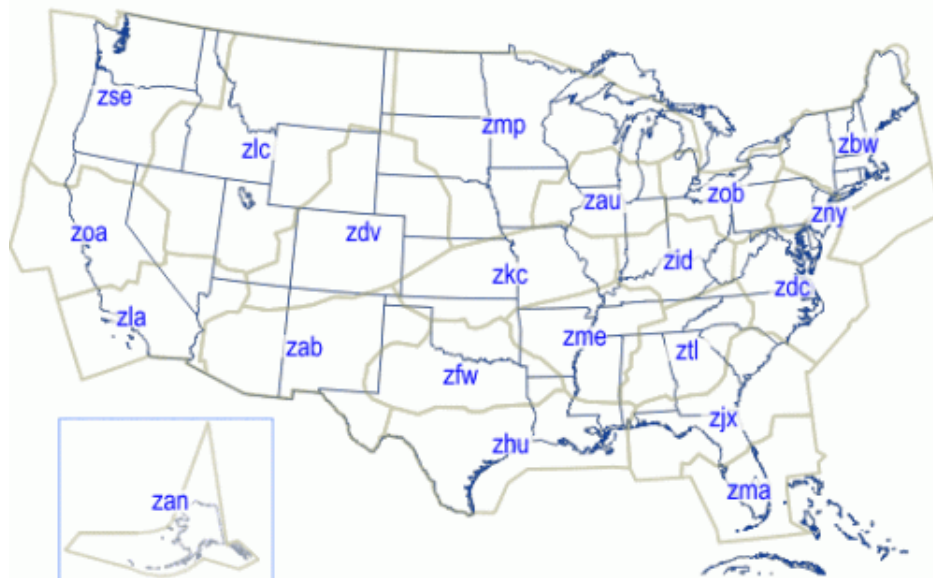


Figura 2.3: Demonstração da Divisão do Espaço Aéreo Americano. Retirado de [Aero, 2016]

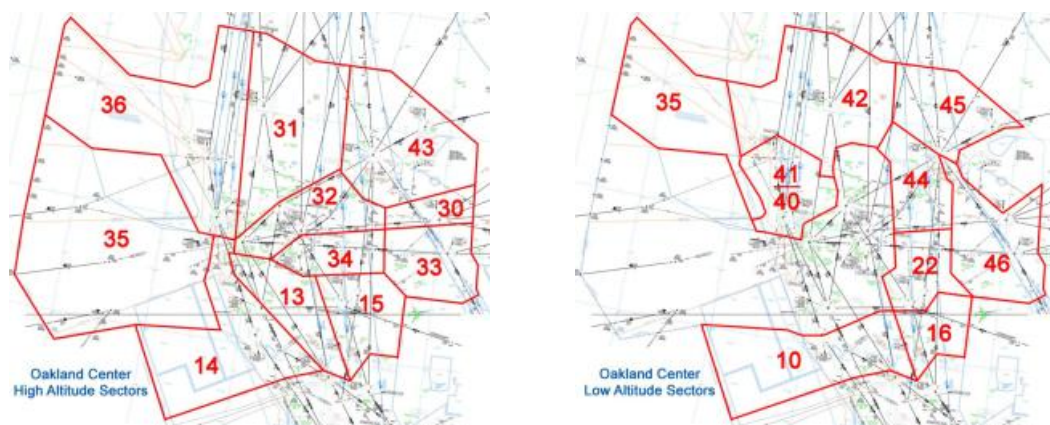
Dependendo da quantidade de tráfego aéreo existente sobre um determinado espaço aéreo, o número de controladores também varia, sendo que os mesmos possuem funções e características

diferentes. Existem três tipos de controladores a trabalharem nestes sectores de forma a garantir a segurança de todos os presentes no espaço aéreo. O primeiro, *Radar controller*, tem como função fazer com que as distâncias de separação *standard* sejam cumpridas, sendo que longitudinalmente, as aeronaves devem estar separadas por 8km caso estejam na mesma altitude e, verticalmente, o controlador deve assegurar que uma aeronave deve estar a uma distância não inferior a 1000ft para uma altitude inferior a 29000ft e não inferior a 2000ft caso contrário. Estes controladores são auxiliados por *Associate controllers* que recebem o plano de voo, 5 a 30 minutos antes de a aeronave entrar no espaço aéreo, sendo possível assim, planear o voo naquele espaço aéreo. Por fim, *Radar Hand-off* tem como função ajudar os outros controladores a analisar o radar quando o tráfego se torna demasiado congestionado e a facilitar a comunicação entre controladores ou entre sectores. É de salientar que os controladores de tráfego aéreo que trabalham nestes edifícios apenas lidam com aeronaves que viajam entre *TRACONs* e que se deslocam em altitudes elevadas, não lidando assim com aterragens e/ou descolagens das aeronaves, visto que o controlo das mesmas é dado aos controladores presentes nas *ATCT* (*Air Traffic Control Tower*) pelos controladores presentes nos *TRACONs*. [FAA, 2006, Nolan, 2010]

2.1.4 TRACON

Denominam-se de *Terminal Radar Approach Control* as instalações pertencentes à *FAA*, onde os controladores monitorizam, através de radares sofisticados, as aeronaves de forma a poderem acompanhá-las durante as várias fases de voo. Estes sectores, que diferenciam em tamanho dependendo da sua localização e aeroportos que servem, podem não ter apenas fronteiras horizontais como também podem possuir fronteiras em altitude, sendo que, por exemplo, o espaço aéreo de *Oakland* está estratificado não só horizontalmente mas como verticalmente, como é visível na Fig. 2.4. Estas instalações têm como principais objetivos ajudar uma aeronave desde que deixa a fase *en route* até que completa a fase de *approach*, e o inverso, ou seja, acompanhar uma aeronave após o *takeoff* até à fase *en route*, onde é entregue ao *Air Route Traffic Control Center*.

Cada um dos sectores visíveis nas figuras possuem uma frequência distinta que é transmitida ao piloto antes do mesmo entrar no espaço aéreo pelo qual o *TRACON* está encarregue. Ao contrário dos homólogos presentes nos *ARTCC* que são responsáveis pelas movimentações das aeronaves em altitude acima dos 10000ft, os controladores presentes nos *TRACONs* estão encarregues de assegurar a segurança das aeronaves em áreas frequentemente congestionadas, quer a aeronave esteja a entrar em fase de *en route* ou para ajudar a aeronave a fazer o *approach* ao aeroporto de destino. Existem 4 tipos de controladores diferentes: *High Altitude descent*, *Low Altitude Descent*, *Approach*, *Feeder* e *Departure Controller*. Após a saída da aeronave da fase *en route*, a mesma passará pelos controladores de alta altitude, baixa altitude, de aproximação até que é entregue aos controladores presentes nas *ATCTs* [Aero, 2016].



(a) Figura Ilustrativa dos Diferentes Sectores de Alta Altitude Existentes no ARTCC de Oakland. Retirado de [Aero, 2016]

(b) Figura Ilustrativa dos Diferentes Sectores de Baixa Altitude Existentes no ARTCC de Oakland. Retirado de [Aero, 2016]

Figura 2.4: Representação das diferenças dos vários sectores existentes.

2.1.5 ATCT

Air Traffic Control Towers são torres existentes nos aeroportos que providenciam auxílio às aeronaves, de forma a garantir a segurança das mesmas e das restantes aeronaves no espaço aéreo, que desejam usar as infra-estruturas dos aeroportos. Para além de estar encarregue de ativar os serviços de emergência quando necessário, os controladores presentes nas ATCTs asseguram, entre outras funções, a inexistência de contratempos na pista, isto é, asseguram que aquando da utilização da pista pela uma aeronave, a mesma está reservada exclusivamente para essa aeronave. São ainda responsáveis, através de comunicação via rádio, de transmitir autorizações às aeronaves, de criar itinerários pelas *taxiway* para que a aeronave as possa usar de modo a sair da pista, e de transmitir informações que facilitem a aterragem, a descolagem aos pilotos. Nos ATCT podem existir até três tipos de controladores, sendo que dois existem obrigatoriamente, *Ground Control*, *Local Control*. O primeiro, *Ground Controller*, tem como sua missão [Nolan, 2010]:

- Dar instruções a aeronaves que estejam em fase de *taxi* e a veículos terrestres de suporte.
- Controlar a iluminação nas *Taxiways*.
- Emissão de autorizações e informação sobre as condições atmosféricas a aeronaves que estejam sob *IFR*
- Receber e retransmitir autorizações para *departure*.
- Em conjunto com o *Local Controller*, coordenar, quando necessário, a passagem de uma aeronave por uma pista ativa.

Devido à necessidade de contacto visual com as pistas, estes controladores trabalham na parte superior da torre que é cercada por vidro. Embora, quase sempre, este tipo de controladores

possua apenas uma frequência rádio para comunicar com os pilotos e com os veículos terrestres, 121.90Hz, existe por vezes a necessidade, de em áreas condicionadas, alargar o espectro das frequências a usar.

Se por um lado, o *Ground Controller* está encarregue de assegurar a separação entre aeronaves depois das mesmas aterram e de controlar as movimentações nas *taxiways* e em qualquer pista inativa, os *Local Controllers* têm como função a separação das aeronaves no espaço aéreo do aeroporto, sendo ainda responsáveis pelas aterragem nas pistas ativas. Este tipo de controlador, entre outras, tem como responsabilidades [Nolan, 2010]:

- Determinar, de acordo com as condições atmosféricas, a pista ativa.
- Transmitir autorizações às aeronaves que desejam aterrar ou levantar voo.
- Controlar a iluminação das pistas.

Embora, e como já foi dito anteriormente, seja mais frequente que existam apenas os dois tipos acima explicados, vários outros tipos podem existir, sendo *Flight Data Controller* um exemplo disso e que tem como principal propósito auxiliar outros controladores nas suas tarefas. [Nolan, 2010]

2.1.6 Plano de Voo

Antes de poderem dar início à sua viagem, as aeronaves que necessitam dos serviços prestados pelo ATC precisam, até 2 horas antes de começar o seu voo, de preencher um plano de voo e enviá-lo para o ARTCC. Como é demonstrado na Fig. 2.5, existem 17 campos que devem ser preenchidos.

U.S. DEPARTMENT OF TRANSPORTATION FEDERAL AVIATION ADMINISTRATION		(FAA USE ONLY) <input type="checkbox"/> PILOT BRIEFING <input type="checkbox"/> VNR <input type="checkbox"/> STOPOVER			TIME STARTED	SPECIALIST INITIALS
FLIGHT PLAN						
1. TYPE VFR IFR DVR	2. AIRCRAFT IDENTIFICATION	3. AIRCRAFT TYPE/ SPECIAL EQUIPMENT	4. TRUE AIRSPEED KTS	5. DEPARTURE POINT	6. DEPARTURE TIME PROPOSED (Z) ACTUAL (Z)	7. CRUISING ALTITUDE
8. ROUTE OF FLIGHT						
9. DESTINATION (Name of airport and city)		10. EST. TIME ENROUTE HOURS MINUTES	11. REMARKS			
12. FUEL ON BOARD HOURS MINUTES		13. ALTERNATE AIRPORT(S)	14. PILOT'S NAME, ADDRESS & TELEPHONE NUMBER & AIRCRAFT HOME BASE			15. NUMBER ABOARD
			17. DESTINATION CONTACT/TELEPHONE (OPTIONAL)			
16. COLOR OF AIRCRAFT		CIVIL AIRCRAFT PILOTS, FAR 91 requires you file an IFR flight plan to operate under instrument flight rules in controlled airspace. Failure to file could result in a civil penalty not to exceed \$1,000 for each violation (Section 901 of the Federal Aviation Act of 1958, as amended). Filing of a VFR flight plan is recommended as a good operating practice. See also Part 99 for requirements concerning DVR flight plans.				

FAA Form 7233-1 (8-82) CLOSE VFR FLIGHT PLAN WITH _____ FSS ON ARRIVAL

Figura 2.5: Exemplo de um plano de voo. Retirado de [Nolan, 2010]

De seguida é explicado cada um dos campos de modo a facilitar assim a compreensão da Fig. [Nolan, 2010]:

1. **Type**

Neste campo deve ser marcado que tipo de voo será feito, determinando assim as regras às quais a aeronave estará sujeita.

2. **Aircraft Identification**

É o local onde deve ser escrito o serial number ou o numero de voo com a Identificação da companhia aérea.

3. **Aircraft type/Special equipment**

Local reservado para que o piloto escreva a identificação de 4 caracteres pela qual a aeronave que voa é conhecida. Deve ainda ser especificado o equipamento de navegação que a aeronave possui. Exemplificando, nesta área um piloto de um *Cesna Citation 650* com *Flight Management System* escreveria *C650/F*.

4. **True Airspeed**

Velocidade à qual o piloto pretende viajar durante o seu percurso.

5. **Departure point**

Representa o local onde a aeronave começa a sua viagem aquando da criação do plano de voo. Normalmente é escrito o aeroporto de origem, no entanto pode ser um *en route fix*. Os pilotos usam 4 caracteres de forma a especificar o local de origem como é recomendado pela *ICAO*. Ao adotar as recomendações feitas pela *ICAO*, a *FAA* adicionar o carácter "K" à designação *IATA*. O aeroporto *JFK (John F. Kennedy)* fica então "KJFK".

6. **Time of Departure**

Horário previsto para que o avião levante voo. Os pilotos indicam este horário em "Zulu". *Zulu* é uma escala de tempo igual em todo o mundo, sendo que é igual ao horário em Lisboa (GMT *Greenwich Mean Time* Lisbon +0) sem correções de tempo.

7. **Cruising Altitude**

Representativo da altitude à qual o piloto pretende voar.

8. **Route of Flight**

Enquanto os pilotos de aeronaves sob *VFR* não são obrigados a navegar de acordo com a rota que escrevem neste local, o cumprimento da mesma é recomendado. Por seu lado, os pilotos sob *IFR* escrevem a rota que desejam fazer, no entanto o *ATC* pode, devido a vários fatores, mudar ou criar uma nova rota diferente. Ao contrário dos pilotos *VFR*, os pilotos *IFR* são obrigados a seguir a rota que lhes é dada. Será visto mais à frente, como estas rotas são escolhidas e escritas.

9. **Destination**

Tal como no *Departure point*, os pilotos devem colocar, de acordo com as recomendações da ICAO, a sigla constituída por 4 caracteres representativa do aeroporto de destino. A outra possibilidade é colocar como destino a cidade e o aeroporto onde se pretende concluir a viagem.

10. ***Estimated time en route***

É desejado que o piloto após estimar a duração do voo, tendo em conta fatores como vento, escreva neste campo a sua estimativa.

11. ***Remarks***

Potenciais anotações que o piloto queira escrever. Este campo não possui qualquer obrigatoriedade.

12. ***Fuel on board***

O piloto deve expressar em horas e minutos, a quantidade de combustível que a aeronave possui. A quantidade de combustível segue um conjunto de regras que deve ser cumprida como é possível ver na secção [2.4.2](#).

13. ***Alternate airport***

Caso seja necessário, o piloto pode colocar aqui os aeroportos de destino alternativos caso haja algum contratempo enquanto se encontra em rota. Tal como em *Routes of Flight*, este campo é apenas uma possibilidade e não uma realidade. Nada obriga o piloto a utilizar estes aeroportos em caso de problemas.

14. ***Pilot's information***

Identificação do piloto como Nome e endereço.

15. ***Number of people on board***

Campo reservado para, como o nome indica, o piloto escrever quantas pessoas irão viajar na aeronave, de forma a auxiliar as equipas de salvamento caso haja algum acidente de aviação.

16. ***Color of the aircraft***

Tal como o campo acima, este apenas serve para ajudar as equipas de salvamento.

17. ***Contact information at destination airport***

Normalmente usado em caso do piloto se ter esquecido de dar como terminado o seu voo como forma de saber se o avião chegou ao seu destino.

2.1.7 NOTAM

De acordo com [Skybrary, 2016] e [Houston, 2016b], um *NOTAM*, abreviação para *Notice to Airmen*, é um documento emitido por autoridades competentes como é o exemplo da *FAA*.

Este documento é emitido com o propósito de informar as mais variadas pessoas e sectores que estão encarregues de controlar ou pilotar o/no espaço aéreo onde irão ocorrer certos eventos que obrigatoriamente mudam a forma como se viaja num determinado ponto. Algumas das razões que levam os *NOTAM* a serem emitidos são: a existência de exercícios militar numa parte do espaço aéreo, pistas ou *taxiways* que estejam inoperacionais, ou que a comunicação via rádio não seja possível. É de realçar ainda a existência de vários tipos deste documento, sendo que podem ter várias nomenclaturas dependendo da sua origem. Na lista abaixo serão esclarecidos alguns dos tipos existentes.

- ***NOTAM(D)*** Reservado para notificações que são partilhadas com alguém que não está localizado dentro da área reservada para o aeroporto. Este tipo de *NOTAM* possui ainda duas distinções, sendo que caso seja precedido por "U", significa que o mesmo não foi validado pela pessoa responsável do aeroporto - *airport manager*, por outro lado, caso o mesmo seja precedido por "O" significa que o *NOTAM* não segue um template *standard*, mas será benéfico para os pilotos a quem é endereçado.
- ***NOTAM(L)*** Apenas disponível para pilotos de aeronaves militares que se encontrem dentro da área destinada ao aeroporto. Ao contrário dos outros tipos, este é emitido por voz.
- ***Center Area NOTAM*** Ao contrário dos anteriores, este possui um alcance de vários aeroportos e, como tal, ao invés de ser transmitido pelo aeroporto responsável, é transmitido pelo ARTCC.
- ***International NOTAM*** Este caso apenas ocorre quando o piloto realiza um voo internacional, sendo que por isso, é usado o formato recomendado pela ICAO. Caso o piloto deseje obter informações *à priori* do seu voo, deve então solicitar este documento, visto que o mesmo não é fornecido sem a sua requisição.

2.1.8 Comunicação ATC-Piloto

Com o crescimento do número de aeronaves que se tem verificado ao longo dos últimos anos, os controladores aéreos têm a seu dispor uma menor quantidade de tempo para comunicarem com as aeronaves que estão a seu controlo. Atualmente a comunicação via rádio segue um ciclo - *ATC: Clearance -> Pilot: Readback -> ATC: Confirm*, que usa a redundância de forma a assegurar que as mensagens foram entendidas de forma correta. Porém caso a mensagem fosse enviada por texto, o controlador conseguiria poupar algum tempo via rádio em que poderia, por exemplo, auxiliar outras aeronaves. É com o propósito de desimpedir o canal via rádio que foi criado o *CPDLC (Controller Pilot Data Link Communications)*. Este mecanismo é usado, actualmente, para trocar mensagens que não são *real time crucial*. Ao contrário da comunicação via rádio o piloto apenas precisa de confirmar a mensagem, não precisando do controlador reconhecer que o piloto recebeu corretamente. [ICAO, 2013, Skybrary, 2015] Sendo responsável por 80% dos acidentes quando não é utilizada corretamente [Airbus, 2017], a comunicação está presente antes

mesmo de a aeronave levantar voo até à mesma chegar ao hangar do aeroporto de destino. Será agora exemplificado algumas das mensagens trocadas entre pilotos e controladores num voo que viaje por diferentes sectores aéreos [Flightgear, 2014]:

1. Taxi Clearance

Esta troca de mensagem tem em vista a obtenção, por parte do piloto, da autorização para executar a manobra de *Taxi*. Este tipo de mensagem segue o seguinte template:

- **Pilot** Station, *callsign*, stand, request taxi
- **Controller** *Callsign*, station, taxi to holding point via *taxiways*
- **Pilot** Taxi to holding point via *taxiways*, *callsign*

2. Takeoff and departure

Através destas mensagens o piloto obtém a indicação para se deslocar para a pista para que esteja pronto a iniciar a partida. É de salientar que a primeira não dá autorização à aeronave para iniciar viagem.

- **Controller** *Callsign*, line up and wait runway number
- **Pilot** Line Up and wait runway number, *callsign*.

Espaço temporal

- **Controller** *Callsign*, wind is heading at speed knots, pista number, cleared for takeoff.
- **Pilot** Cleared for takeoff pista number, *Callsign*

3. Climb, cruise, descent

Após o *takeoff*, o piloto é informado que deve contactar um novo controlador sendo que o mesmo estará responsável em auxiliar a aeronave a circular fora do espaço aéreo local pertencente ao aeroporto. Após receber a frequência do novo controlador, o piloto deve então entrar em contacto e informar as permissões anteriormente obtidas e a sua altitude actual.

- **Pilot** Station, *callsign*, passing *current altitude*, (climbing | descending) to altitude, old clearances
- **Controller** *Callsign*, station, radar contact, squawk, (new clearances se aplicável)

Caso novas autorizações sejam dadas é necessário que haja uma confirmação por parte do piloto. Já em fase *en-route*, o controlador pode informar que a aeronave deve alterar a sua altitude e para tal usa o seguinte template:

- **Controller** *Callsign*, (climb | descend) and maintain new altitude
- **Pilot** (climb | descend) and maintain new altitude, *callsign*

Fundamentos

Por vezes, devido à existência de outras aeronaves na zona, ou devido a condições atmosféricas, por exemplo, é necessário desviar a aeronave horizontalmente. De forma a realizar isto, o controlador contacta a aeronave e transmite a mensagem que se rege pelo seguinte *template*:

- **Controller** *Callsign*, turn (left | right) heading new heading
- **Pilot** Turn (left | right) heading new heading, *callsign*

Por fim, ocasionalmente acontece que o controlador pretenda que a aeronave apenas siga para determinado ponto no espaço aéreo. De modo a informar o piloto da sua decisão, transmite a seguinte mensagem:

- **Controller** *Callsign*, proceed direct *fix/navaid*
- **Pilot** Proceeding direct *fix/navaid*, *callsign*

4. Approach and landing

À chegada das imediações do aeroporto, o piloto muda de controlador outra vez, sendo que este está encarregue de ajudar o piloto a fazer a aproximação ao aeroporto. A troca de mensagens que ocorre é a seguinte:

- **Controller** *Callsign*, cleared for ILS approach runway *number*
- **Pilot** Cleared for approach runway *number*, *callsign*

Após esta troca de mensagens, o piloto pode ser autorizado a aterrar ou ser obrigado a esperar enquanto a pista está ocupada. No caso da primeira opção, o controlador comunica uma autorização ao piloto que segue o seguinte *template*:

- **Controller** *Callsign*, wind is *heading* at *speed* knots, runway *number*, cleared to land
- **Pilot** Cleared to land runway *number*, *callsign*

No caso das condições à aterragem não sejam favoráveis, o controlador pode, por exemplo, pedir ao piloto que execute a manobra de *holding* até que existam condições à aterragem, usando para tal a seguinte mensagem.

- **Controller** *Callsign*, (enter | exit) holding pattern
- **Pilot** (Enter | exit) holding pattern, *callsign*.

5. Troca de controladores

O espaço aéreo, como visto na secção 2.1, é subdividido em vários sectores em que cada sector tem um controlador responsável diferente. É assim necessário que, aquando da saída da aeronave do seu sector, o controlador informe o piloto quem deve contactar de forma a continuar a obter auxílio na sua viagem. Esta informação é trocada pelo seguinte fluxo de mensagens:

- **Controller** *Callsign*, *contact station on frequency*
- **Pilot** *Contact station on frequency, callsign*

2.2 Aeroporto

Um aeroporto é, na sua essência, um conjunto de infra-estruturas capazes de gerir aeronaves, em qualquer fase do voo, que se encontram no seu espaço aéreo. Um aeroporto consiste, portanto, numa área facilmente acedida via aérea e que contenha pelo menos uma pista operacional. Na secção atual serão explorados algumas infra-estruturas e conceitos presentes nos aeroportos.

2.2.1 Pista

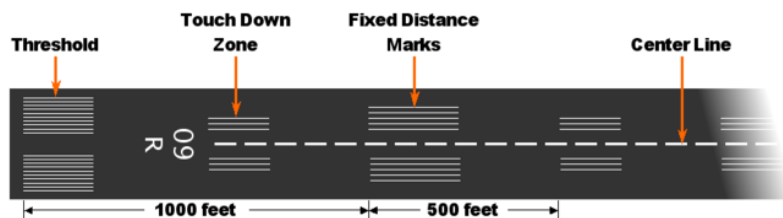


Figura 2.6: Exemplo de uma pista com as respetivas marcações. Imagem retirada de [Wald et al., 2015]

Uma pista é uma pista retangular que está preparada para aterragem e para partida de aeronaves, o que a faz, por isso, numa das partes integrantes mais importantes de um aeroporto. Contrariamente à sua aparente simplicidade, a sua construção é uma tarefa bastante critica, visto que existem vários fatores que influenciam a mesma, como a direção predominante dos ventos. Uma pista é usualmente identificada por um sistema numérico que identifica a direção magnética para a qual está orientada. É ainda de salientar que de forma a implicar a numeração, a *FAA* decidiu arredondar para múltiplos de 10. Na eventualidade de existirem pistas com a mesma numeração é comum a adição de uma letra ("L", "R" ou "C") representativa da posição da pista em relação às outras.

Na Figura 2.6 estão demonstradas algumas marcações existentes numa pista, sendo elas: *threshold*, *Center line*, *Touch Down Zone* e *Fixed Distance Marks*. *Threshold* marca o início da pista onde as aeronaves podem começar a aterrar, *Center line* serve para ajudar o piloto a manter a aeronave no meio da pista. Por sua vez, *Touch Down Zone* demarca o local onde a aeronave deve entrar em contacto com a pista de aterragem. Devido à necessidade de aumentar a capacidade que um determinado aeroporto consegue gerir, a organização americana aprovou 4 tipos de pista, sendo os mesmos explicados abaixo [Wald et al., 2015, chapter C.1].

2.2.1.1 Pista Única

Este tipo de configuração apenas possui uma pista que é partilhada por aeronaves que queiram aterrar ou levantar voo, sendo por este mesmo motivo a configuração mais básica. É construída de

forma a otimizar as ações dos pilotos em relação a fatores como vento, barulho, etc. É estimado que em condições perfeitas, estas pistas consigam acomodar cerca de 60 operações por hora em condições *IFR*.

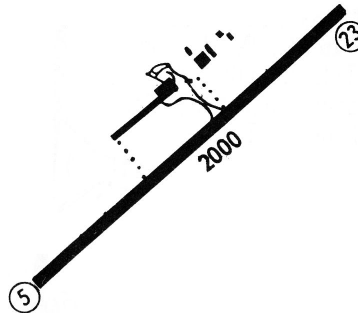


Figura 2.7: Exemplo de uma *single runway*. Imagem retirada de [Freeman, 2016]

2.2.1.2 Pistas Paralelas

É dado o nome de pista paralela a qualquer conjunto de pistas, com pelo menos duas, que possuam a mesma direção geográfica. Esta categoria de pista possui ainda uma distinção por distância entre pistas, podendo as mesmas ser [Wald et al., 2015]:

- ***Closed parallel***

São denominadas de *Closed parallel* quando a distância que separa as diferentes pistas não ultrapassa os 2500ft de distância e existe um mínimo obrigatório de 700ft. Visto que as mesmas não podem operar de forma independente umas das outras, a eficiência é de 50-60 operações por hora em condições *IFR*, sendo assim semelhante à *Single runway*.

- ***Intermediate parallel***

Para obter esta classificação é necessário que as pistas estejam entre 2500ft - 4300ft de distância. Uma das vantagens que este tipo possui em relação às anteriores é a independência entre pistas para diferentes operações, o que leva a um aumento na capacidade das operações para cerca de 60-75 por hora.

- ***Far parallel***

As pistas estão separadas por uma distância mínima de 4300ft o que leva a que ambas possam ser independentes umas da outra qualquer que sejam as operações, o que leva a um aumento na performance que pode variar entre 100-120 operações por hora.

- ***Dual-lane***

Consiste na existência de dois conjuntos de pistas paralelas separadas no mínimo por 4300ft. De forma a aumentar ainda mais a performance, é recomendado a utilização das pistas paralelas mais afastadas do edifício para operações de chegada e reservar para operações de partida o conjunto mais próximo do edifício.

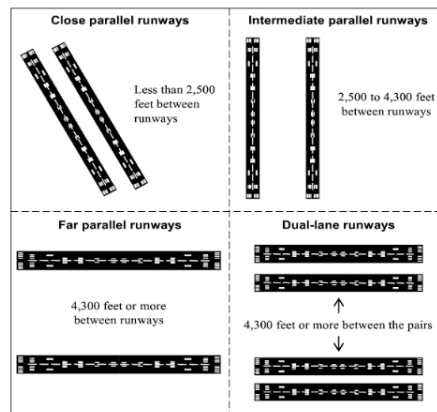
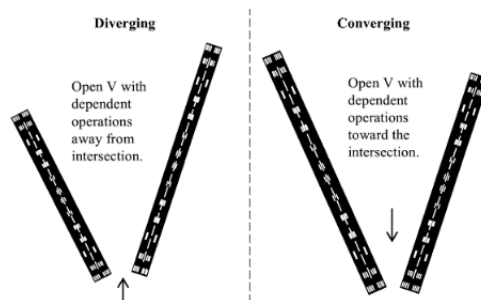


Figura 2.8: Demonstração de possíveis configurações de pistas paralelas. Imagem retirada de [Wald et al., 2015]

2.2.1.3 Pistas em V Aberto

Este tipo de configuração consiste na existência de duas pistas com diferentes orientações que nunca se intercetam, assemelhando-se assim à letra "V". Este tipo de pista é particularmente ótima caso o vento não seja um fator negativo determinante para as operações das aeronaves, no entanto caso haja más condições atmosféricas, apenas uma das pistas será usada. Deve ainda ser salientado que este tipo de pistas podem ser distinguidas ainda por serem divergentes ou convergentes, dependendo da direção das operações.

Figura 2.9: Exemplo de uma configuração *open-v*. Imagem retirada de [Wald et al., 2015]



2.2.1.4 Intersecting runways

Caso exista a ocorrência de ventos fortes de diferentes direções, é recomendada a construção de pistas que se intercetam, de forma a aumentar a performance desse aeroporto. Assim, quando existe prevalência de ventos fortes dificultando as operações numa das pistas, a outra será utilizada. Embora ambas as pistas possam ser utilizadas, caso as condições atmosféricas assim permitam, a sua capacidade depende não só da direção que as operações têm, mas também do local onde as pistas se intersectam. Obtêm-se uma melhor eficiência, caso a intersecção se dê perto do *take-off end*.

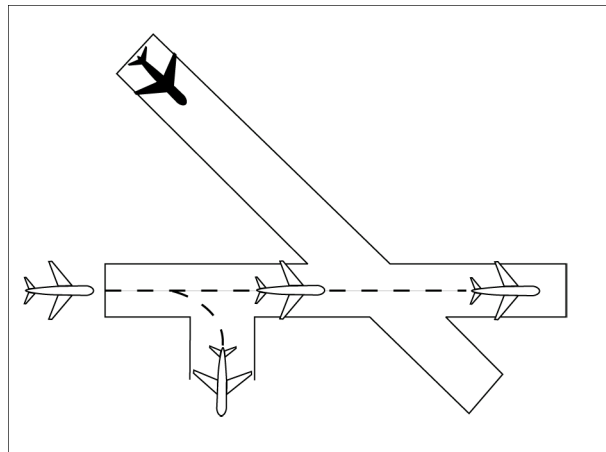


Figura 2.10: Exemplo de uma *Intersecting runways* retirado de [U.S. Department of Transportation Federal Aviation Administration, 2010]

2.2.2 Taxiway

É designado por *Taxiway*, um percurso existente num aeroporto, que faça a ligação entre as pistas e os hangares e terminais, e que é frequentemente utilizado pelas aeronaves antes da sua partida ou depois da sua chegada. É usual serem construídas *express exit taxiway* em aeroportos bastante movimentados, para que aeronaves possam sair das pistas a maiores velocidades, o que permite aumentar a eficiência de operações de um determinado aeroporto. De uma forma geral, as *taxiway* podem ser classificadas pela sua posição ou pela sua direção à pista.

- ***Parallel Taxiway***

É dado este nome às *taxiway* que se estendem paralelamente às pistas.

- ***Entrance Taxiways***

Este tipo de *taxiways* levam uma aeronave até ao início de uma pista para que a aeronave consiga chegar à posição de partida.

- ***Bypass Taxiway***

Taxiways que permitam a determinada aeronave ultrapassar determinada aeronave em direção à sua pista.

- ***Exit Taxiway***

Todas as pistas possuem várias *taxiways* deste género, o que permite a que uma aeronave consiga sair de uma pista assim que a sua velocidade o permitir. São construídas de forma a fazerem entre 30°-45° com a pista.

De forma a se conseguir distinguir entre pistas, são atribuídas letras às mesmas, no entanto, e ao contrário das pistas a mesma pista não pode ter duas marcações iguais, visto que não é necessária a distinção entre os sentidos. Em aeroportos de grande dimensão, as marcações podem usar um sistema alfanumérico de forma a distinguir as várias *taxiways*.

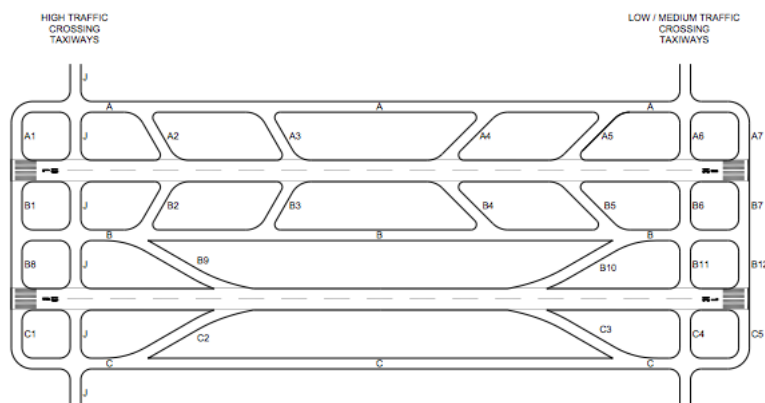


Figura 2.11: Figura ilustrativa de várias *taxiway* retirada de [FAA, 2012]

2.2.3 Cartas Aéreas de Aproximação

Após entrada no espaço aéreo do aeroporto e respetivo pedido de aterragem é atribuída uma pista para a aeronave poder aterrar. De seguida são descritos alguns dos elementos presentes nas cartas de aproximação que servem ao piloto de guia para o procedimento de aterragem (ver Fig. 2.12).

1. Informações sobre a Carta

É através do local representado pelo número 1 que é descrito para que situação deve ser utilizada uma determinada carta aérea. Nesta secção do documento, para além de ser possível encontrar o nome do aeroporto e a respetiva sigla IATA (*International Air Transport Association*), é referenciada a pista de aterragem e o tipo de navegação a ser usado para a aterragem.

2. Informações Sobre Aproximações Falhadas

Todas as cartas de aproximação apresentam dois locais responsáveis pela descrição da rota a percorrer em caso de falha na aproximação/aterragem. Num primeiro local, representado pelo número 2, encontra-se uma descrição textual do procedimento. Por outro lado, de forma a facilitar a captação de dados mais rapidamente, existe uma representação gráfica, representada pelo número 7, do procedimento.

3. Comunicação

Sendo necessário comunicar com os vários tipos de controladores existentes num aeroporto são expostas as várias frequências dos mesmos no local demarcado com o número 3.

4. MSA (*Minimum Sector Altitude*)

Através do que se encontra representado pelo número 4 é transmitida a informação sobre distância e a altitude mínima à qual irá ser sempre assegurada uma distância vertical igual ou superior a 1000 pés. Na carta representada na Figura 2.12, a distância é assegurada acima dos 3100 pés de altitude e num raio de 26 milhas náuticas com centro no início da pista 24R.

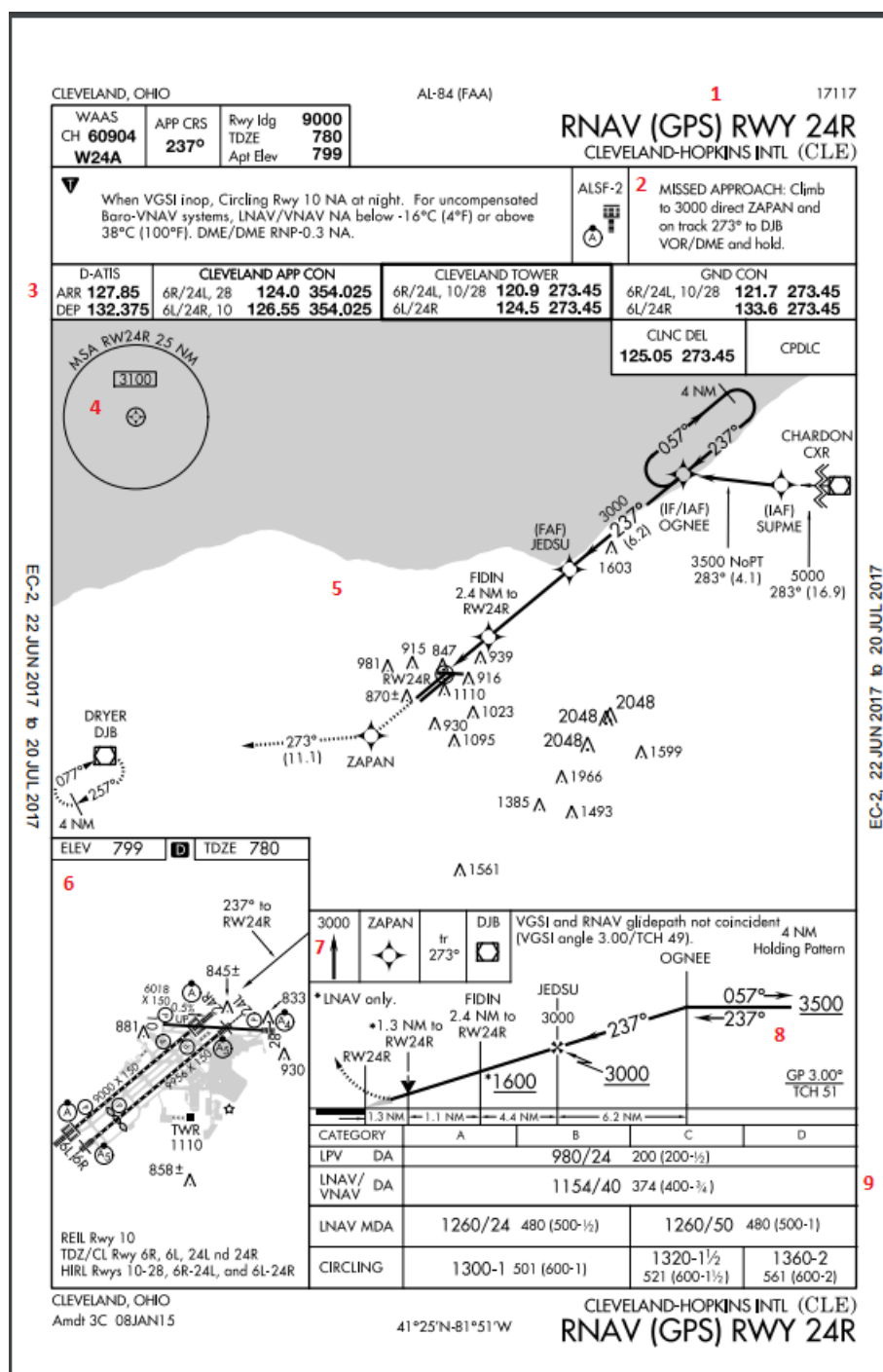


Figura 2.12: Carta de aproximação à pista 6L do aeroporto KCLE, Ohio.

5. Representação gráfica da rota de aproximação

A rota a percorrer e os respetivos mínimos exigidos, para que a aproximação e aterragem sejam concluídas com sucesso, são mostrados graficamente vista de cima e de perfil (ver números 5 e 8, respetivamente). Em ambas consegue-se obter informação relativa à manobra de espera a ser realizada na impossibilidade de iniciar a aproximação e as respetivas altitudes

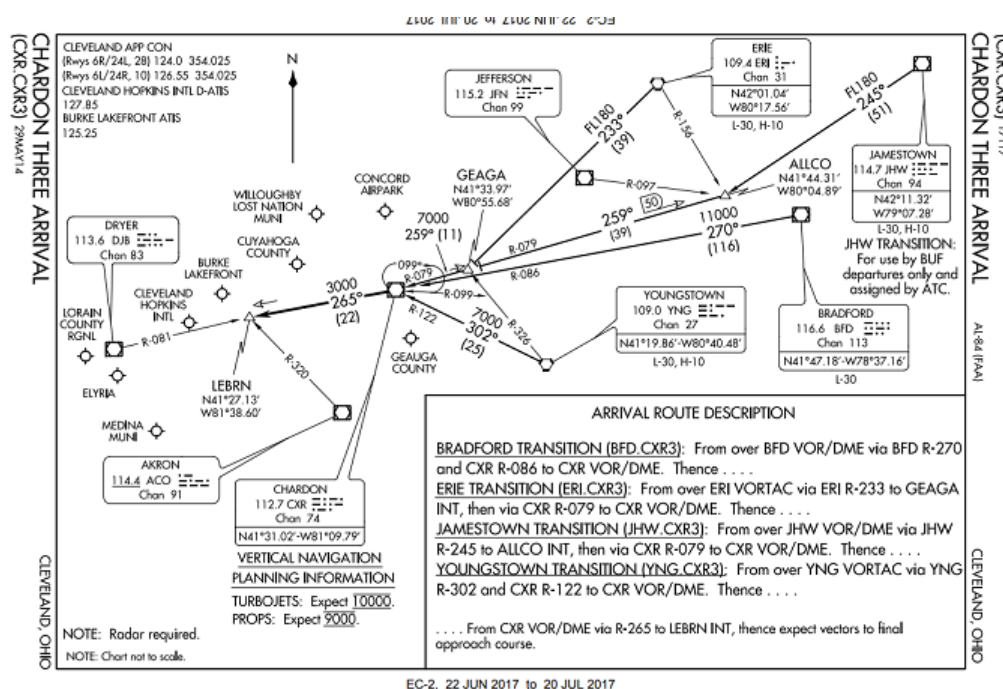
6. Representação do Aeroporto

De forma a que o piloto consiga obter uma informação mais detalhada das infraestruturas do aeroporto o mesmo encontra-se representado na secção catalogada pelo número 6.

7. Mínimos para aterragem

Com o objetivo de informar os pilotos sobre as condições mínimas a serem cumpridas para que seja possível a aterragem foi criada a tabela representada pelo número 9.

Dada a necessidade de manter as aeronaves organizadas são criadas rotas em todas as fases de voo do mesmo. Assim e semelhante à aproximação às pistas dos aeroportos, existem cartas capazes de demonstrar as várias rotas a cumprir para se entrar no espaço aéreo de um aeroporto. Na figura 2.13 está representada uma carta onde é possível obter informações de como se deslocar dentro de uma área reservada para aproximação aos aeroportos.



2.3 Manobra de Espera standard

A manobra de espera, *Holding pattern* na terminologia inglesa, tem como propósito fazer com que determinada aeronave fique relativamente geoestacionária a um determinado ponto no espaço aéreo, enquanto a mesma espera pela permissão para aterrizar. A *FAA (Federal aviation administration)* possui um circuito *standard* em forma de *racetrack* composto por quatro trajetos

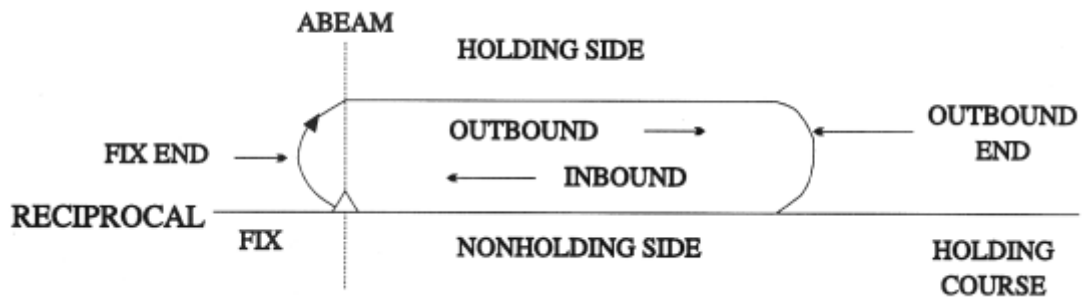


Figura 2.14: Descrição dos pontos referentes à manobra de *Holding*. Retirada de [FAA, 2015a]

- duas curvaturas de 180° e dois que podem possuir durações de 1 minuto ou 1.5 minutos. Alguns dos pontos com relevância são abaixo mencionados [FAA, 2015a]

- **Fixpoint**

Ponto especificado no momento em que a aeronave é redirecionada para a realização do *holding* e pelo qual a aeronave deve sobrevoar de modo a entrar no *Holding Pattern*.

- **Inbound Leg**

Segmento que possui numa das extremidades o *fixpoint*. Uma aeronave deve cronometrar a sua passagem de forma a que mesma, dado a sua altura, demore um minuto ou um minuto e meio.

- **Fix End e Outbound End**

Zona onde a aeronave deve iniciar uma rotação de 3° por segundo no sentido horário, caso do *standard holding*, de forma a entrar na *Outbound leg* ou na *Inbound Leg* respetivamente.

- **Outbound leg**

Parte do circuito que difere do percurso *Inbound* na medida em que pode sofrer alterações com o propósito de corrigir, dado a intensidade do vento por exemplo, a rota de forma a cumprir o tempo preestabelecido para a etapa.

- **ABEAM**

Ponto representativo do momento em que ambas as asas da aeronave encontram-se niveladas, marcando assim o início da etapa de *outbound*.

2.3.1 Entradas para a Manobra de Espera

Atualmente existem três entradas recomendadas, ver na Fig. 2.16, para entrar em manobra de espera, denominando-se por paralela, direta e *teardrop*. Embora qualquer uma possa ser usada, é recomendada a utilização de uma em relação às outras dependendo da direção pela qual a aeronave irá chegar ao ponto de *fix* (ver Fig. 2.15), onde é demonstrado o procedimento recomendado dependendo da direção em que a aeronave se aproxima do ponto *fix* [FAA, 2015a].

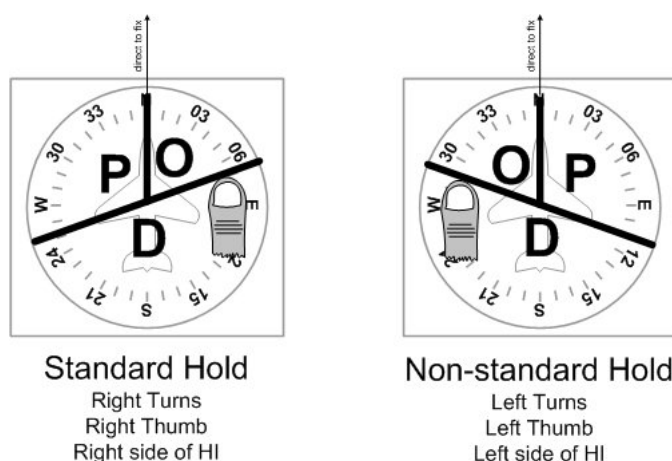


Figura 2.15: Figura ilustrativa de como entrar na manobra de *Holding*. Retirado de [School, 2016]

2.3.1.1 Entrada Paralela

De acordo com a *FAA*, e tendo em conta o padrão de *standard holding*, este procedimento é recomendado quando a aeronave se aproxima do ponto de *fix* pelo sector 1. De forma a realizar este tipo de entrada, a aeronave precisa de sobrevoar até ao ponto de *fix*, sendo que deve continuar paralelamente ao percurso de *inbound*. Após o término do primeiro minuto, a aeronave deve dirigir linearmente até o ponto de *fix* de forma a tomar o curso correto da rota.

2.3.1.2 Teardrop

Neste tipo de entrada, e após sobrevoar o ponto de *fix*, a aeronave deve fazer dirigir-se durante cerca de um minuto numa trajetória que possua não mais de 30° com o percurso de *inbound*. Ao fim deste minuto, deve começar a mudar a rota de forma a intercetar o percurso de *inbound*. Esta manobra é recomendada quando se aproxima do *fix* pelo sector 2.

2.3.1.3 Entrada Direta

Entrada recomendada quando se aproxima pelo sector 3, e é considerada a mais fácil de executar, visto que a aeronave apenas precisa de sobrevoar o *fix* e começa a executar a manobra de *holding*.

2.3.2 Regras para Manobra de Espera pela *FAA*

Com o intuito de assegurar a segurança de todas as aeronaves presentes do espaço aéreo, a *Federal Aviation Administration* possui algumas regras que devem ser cumpridas aquando da realização da manobra de *holding*.

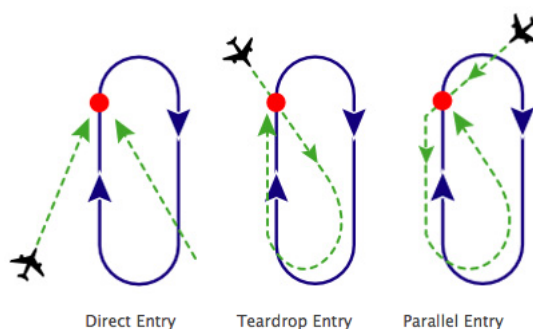


Figura 2.16: Figura ilustrativa dos vários entradas usadas para iniciar a manobra de espera. Retirado de [NashvilleCFI, 2011]

2.3.2.1 Comunicação entre aeronave e controlador ATC

Na eventualidade de não existirem condições favoráveis à aterragem, é da responsabilidade do controlador ATC redirecionar a mesma para um local próprio para a realização da manobra de espera. Sendo recorrente a ocorrência deste tipo de eventos, as cartas dos vários espaços aéreos contêm já a informação sobre os vários pontos de espera (ver Fig.2.17). No caso da manobra documentada representar a manobra que o controlador deseja, o controlador, ao informar o piloto que deve realizar manobras de espera, poderá apenas informar a direção da manobra a ser realizada conjuntamente com "As Published".

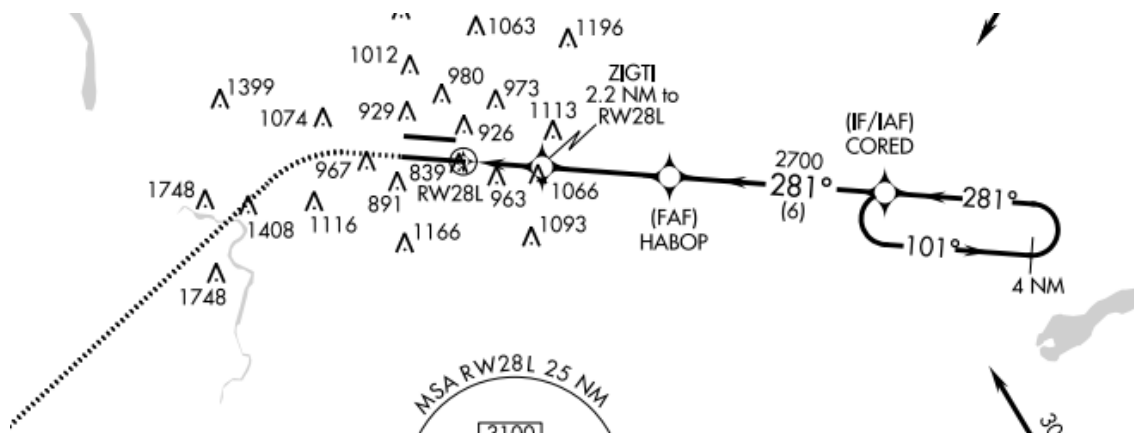


Figura 2.17: Figura ilustrativa de um Local para a Realização de Manobras de Espera.

Por outro lado, para o caso a manobra de espera não se encontrar documentada ou não representar o que é pretendido, o controlador deve assegurar que indica todos os detalhes necessários à realização da mesma: direção da manobra desde do fix, *Leg length* em milhas, *DME (Distance Measuring Equipment)*, ou em minutos, *RNAV (Area Navigation)*, direção das viragens (*standard, ou non standard*, horário previsto para a próxima atualização e, por fim, um *Radial, course, bearing, airway* ou *route* onde é suposto realizar a manobra de espera. Aos pilotos é pedido que reportem o tempo e a altitude/*flight level* a que se encontrará no momento do em que é suposto receberem a atualização por parte do controlador. [FAA, 2015a, chapter 5.3.8]

2.3.2.2 Timing

O tempo que determinada aeronave passa no espaço aéreo enquanto realiza a manobra de espera é controlado com rigor e, como tal, existem certas regras que devem ser cumpridas pelos pilotos. Em relação à *Inbound leg*, a *FAA* estabeleceu que deve ter uma duração de 1 minuto para altitudes inferiores a 14000ft ou 1.5 minutos para altitudes superiores. Por seu lado, a *Outbound leg* não possui um tempo predeterminado à exceção da primeira volta que possui as mesmas regras que a *Inbound Leg*. As seguintes *Outbound Legs* devem ser ajustadas, caso necessário, para garantir o tempo da *Inbound Leg*. O piloto pode, em vez de cronometrar a duração da *inbound leg*, usar uma distância como referência. Neste último caso pode ser usado o sistema *RNAV* que permite ao piloto localizar-se com recurso a múltiplos sistemas de localização (*DME (Distance Measuring Equipment)*, *GPS (Global Positioning System)*). [FAA, 2015a].

2.3.2.3 Velocidade máxima recomendada

As aeronaves estão sujeitas a limites de velocidade, dependendo da altitude a que se encontram, impostos pela *FAA*. Na Tabela 2.2 estão ilustrados os limites para aeronaves civis e militares.

Tabela 2.2: Velocidade Máxima permitida por altitude e tipo das aeronaves

	Altitude(pés)	Velocidade máxima(Nós)
Aeronaves Civis	<6000	200
	<14000	230
	>14000	265
Aeronaves Militares	-	175 - 310

Existem ainda um conjunto de casos que são exceções à regra, sendo eles:

- **Manobras Non-standard** situados entre os 6001ft e os 14000ft onde não é permitido exceder os 210 Nós
- **Manobras de Espera** que estão sujeitos a diferentes limites. Nestes casos, o limite máximo é dito na documentação. Caso os pilotos não consigam executar a manobra de espera com velocidades inferiores deve notificar o *ATC*.
- **USAF(United States Air Force) airfields** possuem limites diferentes, podendo chegar aos 310 Nós.
- **Navy - Marinha Norte-Americana** tem como limite, caso não seja dito o contrário, de 230 Nós.

2.4 Diferenças entre *FAA* e as recomendações da *ICAO*

Dependo da localização do espaço aéreo pela qual a aeronave navega, a mesma é controlada por diversos conjuntos de regras. Nesta secção serão apresentadas algumas das diferenças entre as regras impostas pela *FAA* e as recomendações pela *ICAO*.

2.4.1 Comunicação

A comunicação, no mundo da aviação, é considerada um requisito essencial de forma a garantir a segurança de todas as aeronaves e respetivas vidas humanas. Com o objetivo de minimizar as probabilidades de desentendimentos nas comunicações efetuadas, foram adotadas certas medidas, sendo que uma delas foi a adoção da língua inglesa como *standard*. Tal como a língua portuguesa, a língua inglesa possui várias expressões que são mais facilmente reconhecidas pelos falantes nativos. De forma a solucionar o problema é sugerida pela *ICAO*, a utilização de 'Zero', 'Orbit', 'Vacate' e 'Join' ao invés do usado pela *FAA* 'Oh', '360', 'Clear', 'Enter', respetivamente. Por vezes nem sempre é possível comunicar, seja por questões técnicas ou por questões climatéricas, o que pode porventura gerar graves problemas. Com o intuito de resolver estas situações ambas as organizações possuem regulamentos distintos sendo que a organização americana acredita, em caso de perda de comunicação, que as aeronaves devem viajar pela altitude max(última altitude dada, *minimum en route altitude*), por outro, *ICAO* acredita que aeronave deve manter a última altitude dada. [TVAO, 2010]

2.4.2 Combustível

A organização americana, neste ponto, segue a recomendação da *ICAO* quase por completo [FAA, 2000, page 6], havendo uma diferença no que diz respeito à quantidade mínima de combustível que determinada aeronave deve possuir de forma a assegurar a segurança quer dos tripulantes quer de qualquer aeronave presente no espaço aéreo. Com base no que foi escrito anteriormente, é necessário que seja calculado a quantidade necessária de combustível para a viagem, sendo que para tal, é preciso assegurar que determinada aeronave possui combustível necessário para completar o voo até ao aeroporto de destino (*Trip Fuel*), para voar até 45 minutos a uma velocidade cruzeiro ao chegar ao aeroporto de destino (*Final reserve fuel*), combustível necessário para, caso seja necessário, dirigir para um aeroporto alternativo (*Destination alternate fuel*), combustível para caso de emergência (*Additional fuel*) e por fim, uma quantidade discricionária (*Discretionary fuel*) [ICAO, 2010, chapter 4.3.6.3].

$$\text{Minimum Fuel} = \text{Trip Fuel} + \text{Final reserve fuel} + \text{Destination alternate fuel} + \text{Additional fuel} + \text{Discretionary fuel}$$

Apenas *Discretionary fuel* e *Destination alternate fuel* podem tomar valores nulos, visto que, dependendo das situações podem ou não ser requeridos. O primeiro pois apenas depende do critério do piloto que pode ou não optar por pedir combustível extra para a viagem. O segundo, por outro lado, é onde se encontra a diferença entre o praticado pela *FAA* e o recomendado pela

ICAO. Enquanto a organização americana descreve as condições que devem existir de modo a que haja necessidade de se definir um aeroporto alternativo - caso da existência de neblina abaixo dos 2000ft e que a visibilidade do piloto seja inferior 3 milhas náuticas) [FAA, 2015a], a ICAO deixa ao critério de cada país.

2.4.3 Manobra de Espera

Neste assunto, ambas as organizações estão de acordo em relação à distância vertical entre os vários níveis na pilha de *holding*, 1000ft, e ao tempo que uma volta completa durante a manobra deve demorar, sendo um minuto para casos em que a aeronave voe até 14000ft e 1.5 minutos caso contrário. Um fator que as distingue, contudo, é como as mesmas acreditam que se deve medir o tempo do da manobra, onde a FAA deixa que se escolha entre a *outbound* ou *inbound* enquanto a organização internacional permite a utilização da *Outbound leg* para este efeito ou que a duração da mesma seja dada pela *DME*. Para além desta diferença, as mesmas distinguem-se pelos limites de velocidades a que as aeronaves podem alcançar aquando da manobra de holding. Na Tabela 2.3 são mostradas as diferenças de velocidades entre organizações. [IVAO, 2010]

Tabela 2.3: Limite máximo de velocidades por altura

Altitude (Pés)	FAA (Nós)	ICAO (Nós)
<6000	200	210 Knots
6000 - 14000	230	220 Knots
14000 - 20000	265	240 Knots
20000 - 34000	265	265 Knots
>34000	265	0.8 Mach

Capítulo 3

Revisão da Literatura

Este capítulo tem como principal objetivo a descrição de não só projetos já desenvolvidos cujo tema possui ou está incluído de certa forma no tema da presente dissertação, como também fazer uma introdução às tecnologias a serem usadas como ainda registrar abordagens feitas por outros autores a problemas que assentam nos diversos assuntos a tratar na dissertação. Será então, ao longo do capítulo, abordadas formas parciais ou completas de resolver o problema da congestão de tráfego, envolvendo assim soluções de *ground holding* e *air holding*. Assim comunicação entre agentes, gestão de manobras de *holding*, formas de calcular diferentes rotas de forma a alterar o percurso de uma aeronave em fase *en-route* e estudo de *ground delays* serão estudados neste capítulo.

3.1 Congestão do Espaço Aéreo

A congestão do espaço aéreo tem sido uma constante pelo mundo. O rápido crescimento da necessidade dos transportes aéreos aliado ao facto dos aeroportos já construídos não se conseguirem adaptar de forma a sustentar esse mesmo crescimento leva a um aumento do congestionamento do tráfego aéreo [Zhang and Zhang, 2006]. Assim, de forma a se minimizar prejuízos que em 1991 eram avaliados em 2 mil milhões de dólares americanos [Vranas, 1994], foram estudadas várias formas de conseguir minimizar esse custo. Serão então estudados formas de minimizar o impacto que a congestão do espaço aéreo, nomeadamente *ground holding* e *air holding*.

3.1.1 Planeamento de voo

Devido a fatores, que por escapam do limite do que é passível de ser controlado, como é o caso das condições atmosféricas, a capacidade máxima que um aeroporto pode suportar não é constante ao longo do tempo. Com uma capacidade irregular e com a congestão das aeronaves que já é uma realidade, os autores de [Vranas, 1994] vêem como uma necessidade imprescindível a criação de sistema de apoio à decisão capaz de ajudar os controladores a prevenir potenciais

agravamentos de congestionamentos. Os mesmos defendem que, embora existam vários tipos de abordagens, como melhoramentos nas infraestruturas de aeroportos, uma abordagem a curto prazo é o planeamento de 6 a 12 horas, contendo claro "*ground-holding policies*". Ao contrário do que tinha até então sido feito, exemplo de [Andreatta and Romanin-Jacur, 1987], foi criado um sistema capaz de ter em conta o facto dos aviões estarem quase sempre em viagem, e que o mínimo atraso pode eventualmente ter um efeito dominó nos restantes voos dessa mesma aeronave. Supondo que a capacidade de um aeroporto, embora com diferentes valores para vários intervalos de tempo, é previamente definido, o modelo consegue através da solução de um problema de programação inteira binário, determinar, caso seja possível, o atraso a ser aplicado em terra e no ar ou o cancelamento do voo caso contrário. Sendo obviamente preferível o atraso em terra ao atraso realizado em voo devido aos custos que os últimos possuem, exemplo do custo de combustível e de manutenção, é impreterível o favoritismo ser dado a *ground holding*.

Os únicos sectores tidos em consideração pelos autores [Vranas, 1994] são os do aeroporto de origem e de destino, onde ambos possuem capacidades máximas de chegada e de partida. Para os autores de [Wang et al., 2010] esta é uma das falhas existentes que os mesmos pretendiam melhorar. Foi então criado um modelo matemático com o objetivo de, à semelhança do artigo anteriormente analisado, ser capaz de gerar uma solução capaz de, minimizando o tempo que as várias aeronaves precisam de ser mantidas em terra ou no ar, ter em consideração as capacidades dos vários sectores que as aeronaves podem atravessar. Para além deste primeiro objetivo, os autores desejavam a criação de uma solução capaz de gerar uma solução válida num intervalo de tempo aceitável para ser usada numa situação real. Sendo necessário, aos olhos dos autores de [Wang et al., 2010], algo capaz de gerar uma solução satisfatória num curto espaço de tempo e sendo o problema um *NP problem*, foi criada um algoritmo heurístico.

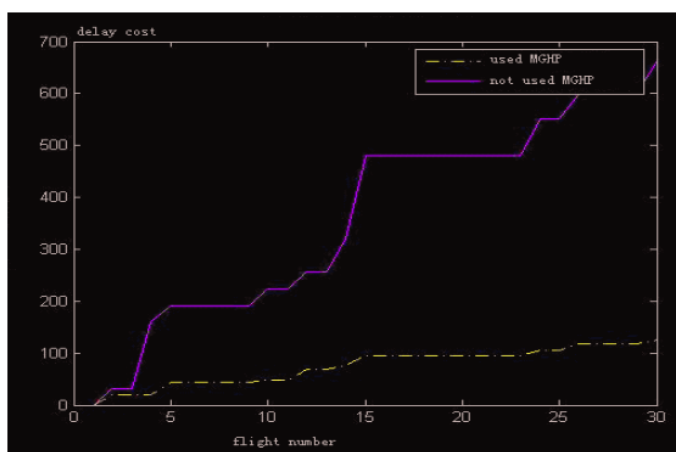


Figura 3.1: Teste efetuado pelos autores do artigo. Imagem retirada de [Wang et al., 2010]

Tal como é possível visualizar através dos resultados mostrados na fig. 3.1, com a adoção do algoritmo o peso dos custos tem tendência a ser inferior.

3.1.2 Otimização de Aterragens

Com o congestionamento aéreo a ser uma realidade atual e com o crescimento desajustado por parte das infraestruturas dos aeroportos tem sido visto um aumento de aeronaves nas proximidades dos aeroportos. Na tentativa de melhorar/impedir o congestionamento aéreo são estudadas várias abordagens relacionadas com a sequenciação das aeronaves de forma a conseguir obter uma melhor taxa de operações (aterragem e partida) para os vários aeroportos.

Em [Salehipour et al., 2013], os autores do artigo referem que o problema de aterragem de aeronaves deve ser abordado através de uma sequencialização ótima das aeronaves e pela divisão das mesmas pelas diferentes pistas disponíveis nos aeroportos. Tendo como objetivo a minimização do desvio temporal entre o horário de aterragem real e pretendido, os autores fazem uso de uma programação inteira de forma a conseguirem sequenciar até 100 aeronaves. No entanto, e reconhecendo a complexidade do problema (*NP-hard*), foi ainda desenvolvido um algoritmo meta-heurístico híbrido de forma a ser possível a obtenção de resultados válidos num intervalo de tempo aceitável para o mundo real. Assim, e tendo em atenção as distâncias obrigatórias entre diferentes tipos de aeronaves, as abordagens desenvolvidas conseguiram obter resultados ótimos para o caso da programação inteira e solução de qualidade elevada para testes com 500 aeronaves.

Ao contrário da abordagem descrita em [Salehipour et al., 2013], os autores do artigo escrito em 2008 [Soomer and Franx, 2008] alteram o fator determinante na sequencialização de aeronaves, (*FCFS - First Come First Served*), de forma a ter em consideração os custos envolvidos em potenciais atrasos de voos para as companhias aéreas. Com a introdução de um método para assegurar que nenhuma companhia aérea seria beneficiada e com um algoritmo heurístico, todas as companhias aéreas obtiveram resultados satisfatórios (custos envolvidos em menor escala). A complexidade temporal do modelo utilizado permite que o mesmo seja utilizado em situações dinâmicas.

Também com o objetivo de melhorar/otimizar a taxa de utilização para ambas as operações existentes, aterragem e partida, os autores de [Balakrishnan and Chandran, 2010] optaram pela introdução de trocas de posição na sequencialização das aeronaves. Apenas pensada para utilização em aproximações a uma única pista, o modelo proposto pelos autores assegura a equidade para as várias companhias aéreas, isto é, uma aeronave apenas pode permutar no máximo para posições que se encontrem a três posições de distância (primeiro para quarto, numa fila *FCFS*, por exemplo). Tal como as abordagens acima descritas, a presente obriga a separação horizontal e vertical das aeronaves de forma a evitar os fenómenos de *wake-vortex*. A presente abordagem tem ainda em consideração possíveis contrapartidas que possam existir em trocas de posições. Os autores concluíram, após a realização dos testes, que é possível sequenciar múltiplas aeronaves com uma complexidade polinomial e como tal a presente abordagem pode ser utilizada em casos reais.

Ao entrar numa das últimas fases de voo, a aproximação ao aeroporto, as aeronaves podem estar entre 8 a 80km de distância do aeroporto, sendo que várias técnicas são usadas de forma a possibilitar à aeronave esperar pela sua permissão para aterrar, como é o uso das manobras de espera ou a redução de velocidade das aeronaves. Contudo esta última é altamente restringida

dependendo do tipo da aeronave, sendo que nem sempre é uma opção capaz de auxiliar a diminuir o congestionamento aéreo. Como solução e forma de diminuir o congestionamento do tráfego aéreo nas proximidades dos aeroportos recorre-se, frequentemente, à utilização de abordagens que fazem uso de um planeamento de aterragem como forma a aumentar o desempenho das pistas de aterragem. Este problema de planeamento a ser usado é conhecido como *Time slot allocation*. Ao contrário dos outros dois métodos utilizados, um planeamento de aterragem não só diminui possíveis atrasos como ajuda a manter a segurança no espaço aéreo. Ao contrário de [Bayen et al., 2004], cuja tentativa de resolução passava pela tentativa de minimização do tempo total de espera de todas as aeronaves ou a minimizar o horário de aterragem da última aeronave, [Martyna, 2013] tem como objetivo a minimização do tempo entre duas aterragens consecutivas tendo ainda em vista a minimização do número de vezes que é necessário recorrer às manobras de espera.

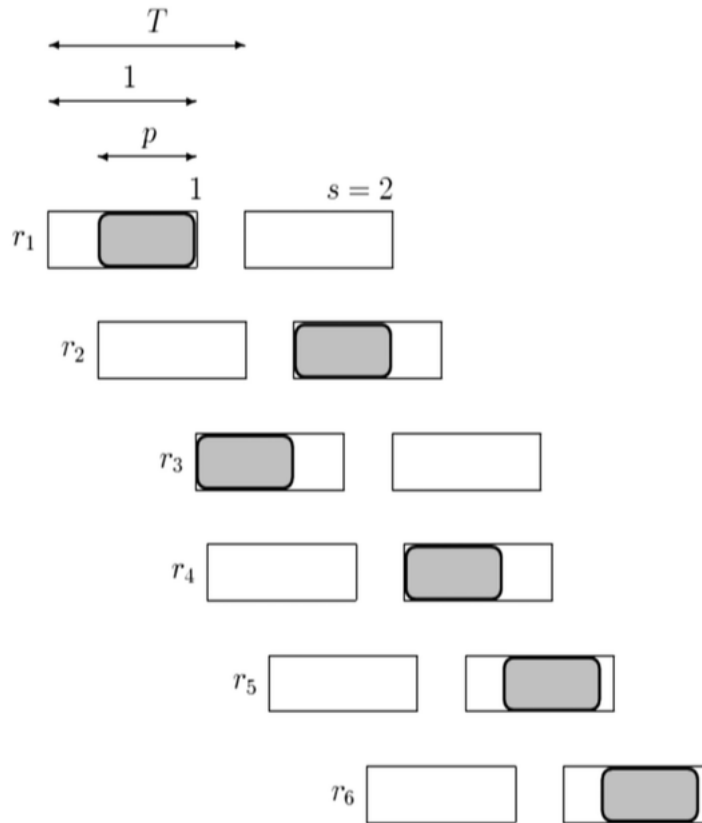


Figura 3.2: Exemplo de um planeamento viável usando 6 aeronaves e 2 janelas. Retirado de [Martyna, 2013]

Na Fig 3.2 está exemplificado uma possível situação onde estão 6 aeronaves à espera de permissão para aterrar, sendo que é dado dois intervalos de tempo para o fazer. Como se pode ver, o parâmetro p ou seja o tempo que uma determinada aeronave têm para aterrar é sempre o mesmo. Tal facto mostra que nenhuma das aeronaves possui prioridade sobre outra, visto que a forma encontrada pela autora do artigo em questão para introduzir prioridades, foi incrementar o parâmetro

p das aeronaves seguintes, ou seja, quanto maior for a prioridade de uma aeronave i , maior será o p para aeronaves $>i$. Esta abordagem obteve bons resultados na medida em que as aeronaves com prioridade esperavam menos tempo que as homólogas sem prioridade.

3.1.3 Otimização e criação de rotas aéreas

O plano de voo, cujo preenchimento é estritamente obrigatório, requer o preenchimento de um campo "*Route of Flight*" onde o piloto anota o trajeto que pretende seguir. Com a sua criação e respetiva validação por parte da entidade responsável, *ATC*, os pilotos podem ver a sua trajetória alterada ocasionalmente ao longo da sua viagem. Embora após a sua criação, a mesma rota possa ser usada múltiplas vezes pela mesma companhia, a sua criação, por outro lado, requer o conciliar de bastantes fatores como condições atmosféricas ou congestionamento previsto ou existente num determinado ponto da sua trajetória. Com o objetivo de conhecer diferentes visões de como lidar com as situações acima mencionadas, estudaram-se várias abordagens desde planeamento de rotas em tempo real, ou criação de um sistema capaz de reduzir atrasos tendo em conta não só condições atmosféricas mas também a saturação de sectores aéreos.

Uma das abordagens mais recentes para a otimização de rotas aéreas é a estudada pelos autores de [Djeumou Fomeni et al., 2017]. Com o objetivo de conseguir otimizar rotas numa ideologia não egoísta, o modelo matemático formulado como um problema de otimização binário faz uso das rotas 4D pré-planeadas dos voos. Estas rotas são depois posteriormente adicionadas ao grafo direcionado representativo do espaço aéreo existente, em que os diferentes nós representam os vários pontos existentes e as arestas as ligações entre esses mesmo pontos. Podendo minimizar a soma dos atrasos existentes, dos custos associados aos desvios das rotas inicialmente propostas ou maximizar a pontualidade das aeronaves, o modelo mostrou-se capaz de retornar rotas que se mostraram melhores às que existiam previamente, no entanto, foi possível perceber a existência de um *trade-off* nos objetivos.

Com o objetivo de criar planos de rotas para múltiplos voos, é projetada uma abordagem que contem 3 passos distintos entre si em [Kiss-Tóth and Takács, 2014]. Num primeiro passo e com o objetivo de impedir que as aeronaves sobrevoassem zonas do espaço aéreo proibidas, os autores utilizaram o algoritmo de *Dijkstra* para a criação de rotas para os vários voos a simular. Contendo ainda apenas uma dimensão bidimensional e de forma a melhorar a performance das rotas, os autores recorrem a algoritmos genéticos de forma a descobrir variações das rotas inicialmente criadas e que possam utilizar a direção dos ventos a seu favor. Após fixar a rota 2D, são então adicionadas as variáveis de altitude e velocidade das aeronaves, sendo que se tenta minimizar as oscilações das mesmas de forma a se conseguir também minimizar o custo global envolvido.

À semelhança das abordagens acima descritas na presente secção, [Djeumou Fomeni et al., 2017] e [Kiss-Tóth and Takács, 2014], [Zillies, 2013] tenta-se otimizar as várias rotas 4D dos múltiplos voos existentes na simulação. Nesta abordagem e na tentativa de criar rotas que evitam áreas afetadas por condições atmosféricas adversas, os autores dividem em duas instâncias o problema, sendo que numa primeira fase, são otimizadas as rotas dos vários voos sem ter em consideração

as outras aeronaves e numa segunda instância, é então assegurada a distância das várias aeronaves em todo o seu percurso através de um método heurístico baseado em relaxação Lagrangiana.

Os autores de [Chaimatanan et al., 2015] abordam não a otimização de rotas 4D já existentes mas a criação dessas mesmas rotas. Possuindo para cada voo diferentes rotas passíveis de serem utilizadas, os diferentes níveis de voo possíveis e os vários horários de partida aceitáveis do aeroporto de origem, esta abordagem faz uso da programação inteira para separar as aeronaves em todos os momentos do seu voo. A presente abordagem utiliza manobras como adiamento da partida do aeroporto de origem, alteração de níveis de voo ou criação de pontos do espaço aéreo fictícios para alongar a rota de forma a que consiga resolver potenciais conflitos de rotas.

3.1.3.1 *Robust Algorithm for Real-Time Route Planning*

Por vezes, sejam os voos de origem militar ou comercial, existe a necessidade de se poder corrigir a trajetória outrora definida e aceite pelo ATC. Segundo os autores de [Szczerba et al., 2000], os algoritmos em uso para o planeamento de rotas aéreas apenas fornecem uma rota que apenas tem em consideração uma função de custo predeterminada. Dito isto, os autores propõem a criação de um algoritmo capaz de gerar, em tempo real, uma rota que tem em atenção múltiplos parâmetros como a dimensão da rota desejada, ângulo máximo de rotação, distância mínima a percorrer até ser permitido uma rotação e, por fim, o ângulo pelo qual a aeronave deve chegar ao seu destino.

- **Dimensão da rota:** *Route Distance Constraint*

Por vezes, é necessário encurtar a dimensão do caminho a percorrer devido a problemas técnicos nos aparelhos das aeronaves, questões relacionadas com o consumo de combustível, ou por questões de foro temporal. Assim esta variável é usada para assegurar que a futura trajetória terá uma distância igual ou inferior à atual.

- **Ângulo máximo de rotação:** *Maximum Turning Angle*

Esta restrição serve para garantir que a aeronave, durante a sua nova trajetória nunca irá realizar uma rotação com um ângulo superior a um valor predefinido. Os autores salientam ainda a importância desta restrição em, por exemplo, voos em formação, onde uma mudança de direção brusca pode aumentar a possibilidade de uma colisão.

- **Distância Mínima a percorrer até ser permitida uma rotação:** *Route Distance Constraint*

De acordo com Lizzie Dearden [Dearden, 2015], a causa dos acidentes de aviação é, na sua maioria, causada por erros dos pilotos. Sendo a fadiga acumulada um dos fatores que contribui imensamente para o acontecimento de erros, é absolutamente necessário a criação de procedimentos que permita ao piloto não se desgastar. Esta restrição, assentando na premissa que em viagens longas a constante muda de direção provoca fadiga e aumenta os erros dos pilotos e tendo como objetivo a minimização da fadiga dos mesmos, define uma distância mínima a percorrer em linha reta antes que seja possível a alteração de direção.

- **Ângulo final com o destino: *Route Distance Constraint***

Aquando da aproximação ao aeroporto por parte de uma aeronave, é-lhe dada uma pista disponível para executar a manobra de aterragem. Assim, de forma a facilitar esta manobra, esta variável pretende assegurar que a aeronave atinja o seu destino com determinado vetor direcional.

É visto como uma necessidade emergente a possibilidade de gerar novas trajetórias em tempo real. Sendo, segundo os escritores do artigo, a geração de uma rota ótima um *NP-Problem* onde a memória e tempo requeridos seriam realisticamente impossíveis para uma aplicação que necessita de ser capaz de gerar uma resposta num curto espaço temporal. Com a necessidade de resolver o problema de geração de uma rota num espaço de tempo e com um uso de memória razoáveis, os autores criaram um algoritmo que se distancia dos restantes por ser *grid-based* e *graph-based*. Com esta junção, foi possível de maximizar as vantagens e minimizar as desvantagens de ambas abordagens. O algoritmo faz uso de uma tabela $M \times N$ onde é marcada o início e o destino desejado e onde cada célula corresponde a uma localização. A cada célula está ainda associada um valor representativo do custo inerente à viagem por essa célula calculado numa primeira etapa do algoritmo (*cost estimation step*). O conjunto destes valores será armazenado numa lista bidimensional $M \times N$ (MC - map cost). Após estimar os custos, *cost minimization step* recebe MC e para cada célula, descobre o custo mínimo BC - *best cost* de forma a se atingir, da célula atual, o destino.

	12 50	10 38	16 28	12 12	14 0	Goal
	84 122	272 300	275 287	15 15	15 15	
	236 358	45 167	210 225	13 28	13 28	
	32 199	236 403	299 327	222 250	214 242	
Start	23 222	23 222	25 247	34 276	24 266	

MC Value
BC Value

Figura 3.3: Exemplo dado pelos autores de uma tabela gerado pelo algoritmo em fases iniciais. Retirado de [Szczerba et al., 2000].

Segundo [Szczerba et al., 2000], a maioria dos algoritmos atualmente usados, fazem uso do algoritmo A* (*A star*) como forma de explorar as possibilidades com o intuito de encontrar uma solução para o problema existente. Contudo, segundo os mesmos, de forma a encontrar a solução ótima, o algoritmo pode incorrer num uso exponencial de tempo e memória. Como exemplo, é dito que, no caso da restrição que define o ângulo máximo de rotação, o algoritmo A* poderia obter um número infinito de nós. Os autores sugerem então um algoritmo, SAS (*Sparse A* Search*) que se mostra capaz de convergir para um solução aceitável em tempo-real, introduzindo para isso o

correr preciso de nós. De seguida é mostrado de que forma cada restrição é incorporada e tida em conta na abordagem criada por [Szczerba et al., 2000].

- **Minimum Route Leg Length and Maximum Turn Angle**

Assumindo que se tem como *input* a localização inicial e final assim como as restrições que estão a ser descritas, existe a possibilidade de algumas posições serem impossíveis de alcançar. O algoritmo divide então estas posições em sectores. É de realçar a importância do número destes sectores, visto que, embora o uso de memória aumente, a qualidade do *output* é melhor. Após a divisão em sectores das células às quais se pode atingir, é escolhida a célula, para cada sector, que possui o *minimum cost* e o mesmo é colocada num *min-heap*. O *minimum cost* é o valor calculado pela função heurística, onde $g(x)$ assume o valor do custo desde a posição inicial até a posição atual enquanto $h(x)$ é o valor que se estima custar a viagem da posição atual até ao destino.

- **Route Distance Constraint**

A introdução deste *input* ao algoritmo tem em vista impedir que a rota a gerar, desde o nó inicial (ponto inicial) até ao nó destino (ponto final), possua uma distância a percorrer nunca superior ao valor do argumento. Esta restrição vem resolver possíveis problemas onde o combustível é escasso ou, por exemplo, é necessário que se chegue ao destino mais rapidamente. Para a introdução desta restrição no algoritmo, os autores de [Szczerba et al., 2000] limitam a introdução de certos nós no *min-heap* do SAS. Assim apenas nós capazes de cumprir a regra $D(x) + SL(x) \leq d-max$, onde $D(x)$ é a distância que já se viajou até ao nó x e $SL(x)$ é a distância em linha reta (distância mais curta) do nó x até ao nó final, são expandidos.

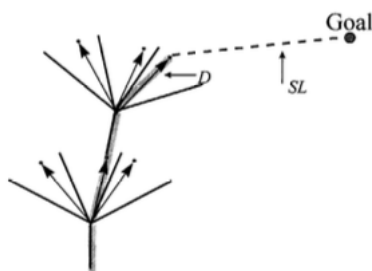


Figura 3.4: Exemplo de uma extensão de nó dada pelos autores. Retirado de [Szczerba et al., 2000].

Na Figura 3.4 é possível ver um exemplo, onde apenas um dos nós é expandido visto que a regra acima é cumprida.

- **Fixed Approach Vector to Goal Position**

De forma a garantir que uma aeronave chegue ao seu destino por um certo ângulo, os criadores do algoritmo tiveram a ideia de aumentar o possível custo das áreas que envolvem o destino. Na Figura 3.5 é demonstrado as áreas sobre as quais se devem aumentar o custo.

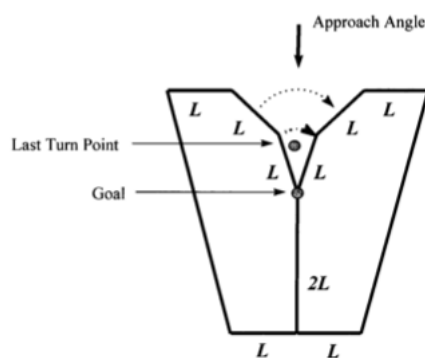


Figura 3.5: Exemplo de um *Bucket-shaped area* dada pelos autores. Retirado de [Szczerba et al., 2000].

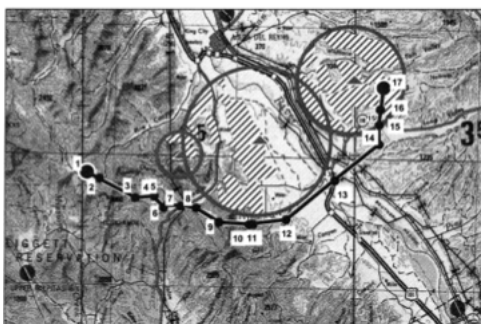
O algoritmo desenvolvido e descrito no artigo [Szczerba et al., 2000], foi testado com dados reais sob o espaço aéreo californiano. De forma a conseguir testar todas as restrições implementadas, os autores do artigo para além de optarem por uma região onde existem ameaças à segurança da aeronave, realizaram testes com diferentes valores máximos de distâncias a percorrer assim como a existência ou não de um ângulo de aproximação. De forma a melhor entender os resultados presentes nas Figuras 3.6a, 3.6b e 3.6c, é necessário explicar que os três círculos visíveis representam áreas de perigo que são perceptíveis e que as áreas sublinhadas representam áreas com um grau de perigosidade mais elevado. Num primeiro caso (fig. 3.6a), o algoritmo foi executado com uma *Route Distance Constraint* de valor igual a duas vezes e meia à distância que seria necessário à aeronave para chegar ao destino case viajasse em linha reta. Com este valor como restrição, os autores verificaram que o algoritmo gerava uma rota o mais segura possível, isto é, mais longe possível das zonas que implicam perigo e numa zona onde o terreno não tem grande relevo. Por outro lado, ao diminuir a restrição para 1.3 pontos é possível notar que a rota se aproximou da zona de perigo embora que não entre num espaço aéreo demarcado como perigoso. Por fim, ao introduzir a restrição de *Fixed Approach Vector to Goal Position* e mantendo a restrição do tamanho da rota em 1.3 pontos, a rota continua a cercar as zonas demarcadas como perigosas. É de salientar o facto da rota, após entrar numa zona perigosa, continua a evitar as zonas marcadas por sublinhado (visível nos pontos 9-10-11).

3.1.3.2 Geração e rotas tendo em conta as possíveis condições atmosféricas adversas

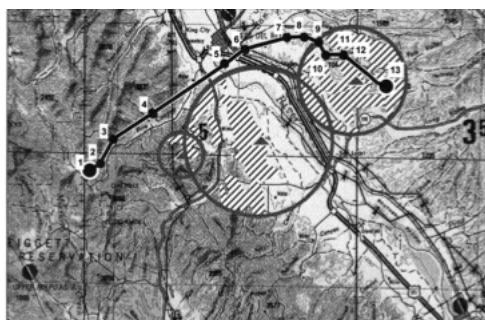
Autores de artigos como "*Design and Evaluation of a Dynamic Programming Flight Routing Algorithm Using the Convective Weather Avoidance Model*"(2009) [Ng et al., 2009] e "*Identification of Robust Routes using Convective Weather Forecasts*"(2009) [Pfeil and Balakrishnan, 2012] vêm as condições atmosféricas como um factor capaz de causar prejuízos alargados às companhias aéreas e ao estado americano. Este último com base em *Your flight has been delayed again* (2008) [U.S Senate Joint Economic Committee, 2008] admite que os atrasos ocorridos no país norte-americano custaram cerca de 47 mil milhões de dólares americanos assim como a percentagem (25 pontos percentuais) de voos atrasados deve-se às condições atmosféricas.



(a) 1º resultado apresentado pelos autores. Retirado de [Szczurba et al., 2000].



(b) 2º resultado apresentado pelos autores. Retirado de [Szczurba et al., 2000].



(c) 3º resultado apresentado pelos autores. Retirado de [Szczurba et al., 2000].

Com base na premissa de que é necessário um sistema capaz de otimizar o rendimento e a eficiência do atual sistema de controlo norte-americano, é proposto o desenvolvimento de um algoritmo capaz de gerar uma rota de voo ótima tendo em conta as condições atmosféricas assim como congestionamento do espaço aéreo. Os autores de [Ng et al., 2009] pretendem que o algoritmo seja visto como uma alternativa as rotas constantes que são usadas pela *FAA*. Embora algumas abordagens conhecidas para este problema façam uso do algoritmo A* (A star) [Hart et al., 1972] e D* (D star - versão aprimorada do A*) [Stentz, 1995], [Ng et al., 2009], por outro lado, decidem enveredar pelo uso de programação dinâmica, que segundo os mesmos, para além de ser capaz de gerar uma solução globalmente ótima caso exista uma solução possível, a mesma solução pode ser obtida em tempo linear $O(n)$ onde n representa o numero de arestas entre nós.

1. Numa primeira fase, o algoritmo transforma a posição inicial da aeronave e a posição desejada (latitude e longitude) em coordenadas Cartesianas. Com isto definido, o utilizador, ao indicar o a dimensão das células, o número de estados e de estágios podem ser definidos.
2. Numa segunda fase, é necessário estimar para cada estado X_{ij} o tempo de chegada. Os autores, de forma a simplificar, ao invés do cálculo do caminho mais curto e mais longo para T_{min} e T_{max} apenas multiplicam uma constante pelo T_{min} de forma a deduzirem o T_{max} . O T_{min} , por seu lado, é calculado pela distância que separa os dois estados na vida real, tendo em conta a curvatura da Terra *great circle distance*, e pela velocidade máxima da aeronave em questão.

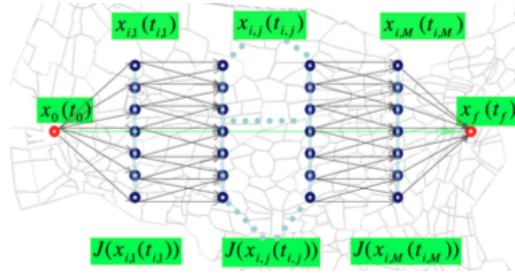


Figura 3.7: Exemplo dado pelos autores de uma *grid* com estados (x) e estágios (J). Retirado de [Ng et al., 2009]

3. Ao chegar a este fase, o algoritmo, com a finalidade de poupar memória e tempo, apela à criação de algumas regras com o intuito de diminuir a quantidade de estados que são alcançáveis a partir do estado atual. Para isso, recorre-se à lógica, onde caso a aeronave apenas pode, no máximo, escolher três estados, no caso de não estar nas fronteiras da *grid*, ou dois estados caso contrário. Assim, através destas regras, está implícito que a aeronave não pode retroceder no espaço. É possível ver as conexões possíveis na Fig.3.7.
4. Terminadas as fases anteriores, o algoritmo calcula o custo entre as várias conexões possíveis entre estados. Como é possível ver na Equação 3.1, o custo será uma combinação de combustível usado, existência de condições atmosféricas adversas e a possível congestão da área pela qual a aeronave terá de passar. De forma a calcular o custo do combustível apenas se multiplica a distância percorrida até ao estado atual por uma constante D_{fuel} . O custo referente à congestão no espaço pode ser atribuído diferentes valores dependendo da quantidade de tráfego que se prevê existir na hora prevista de chegada da aeronave àquele sector, sendo que caso o sector esteja na sua capacidade máxima, o respetivo custo deve ser definido como infinito. Por fim, o custo referente às condições atmosféricas é demonstrado pela Equação 3.2, sendo que o seu valor, à semelhança do custo de congestionamento, deve ser infinito em caso de existência de condições atmosféricas adversas.

$$J(x_{i,j}(t_{i,j})) = \min_{t'} [D_{i,j,t_{i,j}}^{i',j+1,t_{i',j+1}} + W_{i,j,t_{i,j}}^{i',j+1,t_{i',j+1}} + C_{i,j,t_{i,j}}^{i',j+1,t_{i',j+1}} + J(x_{i',j+1}(t_{i',j+1})))] \quad (3.1)$$

5. Concluídos os cálculos de custo para cada estado admissível, é calculado o custo ótimo para cada estado através da função mostrada na Equação.3.1.
6. Por fim, a rota ótima é calculada minimizando todo o custo total desde a origem até ao destino.

Este algoritmo, com o objetivo de calcular a probabilidade de um piloto desviar a sua rota devido às condições atmosféricas, faz uso de *CWAM* (*Convective Weather Avoidance Model*), um modelo que integra as previsões meteorológicas oriundas de *CIWS* (*Corridor Integrated Weather System*) com as decisões dos pilotos para que se possa prever quando um piloto irá desviar a

sua trajetória devido a existência de condições atmosféricas adversas. Com esta probabilidade, é possível calcular o *deviation cost* usado no algoritmo no passo 4, sendo que na formula é assumido que $K_{deviation}$ é uma constante definida pelo utilizador.

$$\begin{aligned} W_{i,j,t_{i,j}}^{i',j+1,t_{i',j+1}} &= E[K_{deviation}] \\ &= K_{deviation} * P(\mathcal{D}_{[t_{i,j},t_{i',j+1}]}) \end{aligned} \quad (3.2)$$

De forma a testar a sua abordagem, as rotas geradas foram comparadas com as rotas já pre-determinadas pela *FAA*. Considerado como ótima, a rota a 35000 pés de altitude mais curta e que cuja probabilidade de desvio é mais reduzida, foram testadas as rotas para cerca de 50 voos.

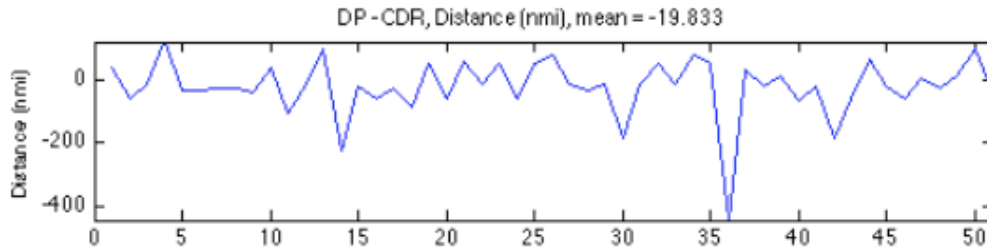


Figura 3.8: Diferença entre distâncias das rotas geradas pelo algoritmo e as rotas pré-definidas pela *FAA*. Retirado de [Ng et al., 2009]

Como é possível ver na fig.3.8, quando comparadas as rotas geradas pela abordagem dos autores, as rotas são em média 3% mais curtas que as rotas ótimas criadas pela *FAA*. Para além de conseguirem gerar rotas, à luz da definição dos autores, ótimas o algoritmo é ainda linear em questões de tempo.

Se por um lado, os autores do artigo acima descrito, criaram um algoritmo capaz de gerar, com base em informação probabilísticas, rotas ótimas capazes de evitar condições desfavoráveis ao longo da sua viagem completa, [Pfeil and Balakrishnan, 2012] determinam rotas, com o intuito de descobrir a melhor rota possível com base nos dados meteorológicos, em territórios onde as aeronaves aproximam-se dos aeroportos. Na fig. 3.9 é visível a forma como é retratado o problema: duas circunferências demarcam a área a estudar, sendo que *C0* demarca a área em torno de um aeroporto com um raio de 75km enquanto, *c1* define a área envolvente ao aeroporto com um raio de 10km. As setas a branco definem os possíveis ângulos de entrada para aterragem em *Hartsfield Atlanta International airport* já no *final approach (c1)*. A vermelho estão demarcadas as zonas onde está previsto as condições atmosféricas adversas à aviação.

Sabendo da imprevisibilidade das condições atmosféricas, ver fig. 3.10, os autores preferem o uso de um serviço que, em intervalos de 5 minutos, vai atualizando as suas previsões. Estas previsões são fornecidas através de uma tabela em que cada célula representa uma área com $1m^2$. Estas áreas possuem um valor, podendo variar de 0 a 255, que posteriormente são convertidos para uma escala com 7[0-6] níveis diferentes, sendo que acima de 3, a célula em consideração possui condições atmosféricas adversas para o piloto.

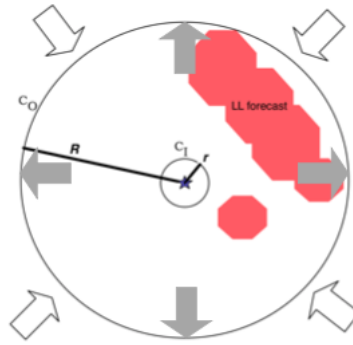


Figura 3.9: Exemplo dado pelos autores de uma área onde são estudadas as rotas. Retirado de [Pfeil and Balakrishnan, 2012]

Como é possível visualizar na fig.3.11, as condições atmosféricas previstas compensam a progressão da aeronave. [Pfeil and Balakrishnan, 2012] supõe que a aeronave chega à fronteira de $C0$ no momento $t0$. É de realçar que esta abordagem faz uso dos vários intervalos de previsões de forma a compensar o movimento da aeronave, que se supõe movimentar a 180 Knots .

De forma a determinar a melhor rota possível são criadas n rotas possíveis com início em $C0$ e com final em $C1$ (locais visíveis na Fig. 3.9). A ideia passa por usar um algoritmo de caminho mais curto de forma a produzir a rota. Com o propósito de compensar possíveis erros de previsão, uma rota, de forma a ser considerada ótima, necessita de não interceder qualquer célula com um nível igual ou superior a 3 e de não existir num raio de $b \text{ km}$ nenhuma célula com condições atmosféricas adversas previstas. O problema de caminho mais curto, é então resolvido pela construção de um grafo dirigido onde os nós representam constituem todos os pixeis que estão à distância de $B \text{ km}$ da rota originalmente traçada.

3.2 Otimização do espaço aéreo

De acordo com *Joint Planning and Development Office (2007)* [NextGen, 2007] é expectado que, até 2027, o tráfego aéreo existente no espaço aéreo norte-americano possa duplicar ou mesmo triplicar face aos valores existentes em 2007. É então necessário, segundo *Shon Grabbe and Banavar Sridhar (2009)* [Grabbe et al., 2009], o desenvolvimento de processos capazes de aumentar significativamente a eficácia do *NAS (National Airspace System)*. A abordagem sugerida pelos autores do estudo revela-se como uma otimização sequencial capaz de integrar um modelo de controlo de descolagens de aeronaves com um sistema de simulação. O modelo de controlo de descolagens, modelo determinístico de programação inteira, tem como principal objetivo o atrasar um ou mais voos caso a capacidade do espaço aéreo do aeroporto assim o justifique. Como solução, para se reduzir a complexidade do modelo de descolagens, apenas atrasos são definidos no modelo acima referenciado, deixando para a parte da simulação a resolução de problemas que possam surgir numa fase da viagem após a descolagem.

Na fig.3.12 está definida a arquitetura otimização-simulação desenhado em [Grabbe et al., 2009]. O sistema de simulação *Primary Simulation*, sistema escolhido pela NASA para ser *Future ATM*

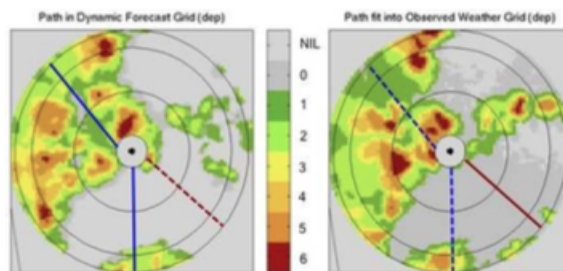


Figura 3.10: Exemplo dado pelos autores de condições previstas Vs condições reais. Retirado de [Pfeil and Balakrishnan, 2012]

Concepts Evaluation Tool (FACET), recebe como *input* os horários dos voos e os seus respectivos planos de voo, informação acerca das condições meteorológicas e dados relativos à adaptação do espaço aéreo. O modelo sugerido pelos autores, contempla cerca de três procedimentos distintos, estando os mesmos coloridos com cores distintas na fig.3.12:

1. Com cor verde, informação como localização e velocidade $x(t)$ de todas as aeronaves existentes na simulação, e informação relativa às condições atmosféricas $w(t)$ são usadas, hora a hora, com o intuito de definir, caso necessário, atrasos em voos que ainda se encontram em terra. Numa primeira instância, são calculadas as posições previstas das aeronaves num futuro próximo. Após calculadas, o modelo calcula de que forma as condições atmosféricas irão afetar a capacidade de determinado aeroporto. Por fim, com os dados já calculados o modelo consegue definir atrasos em voos ainda por levantar de acordo com a capacidade prevista. O modelo "*Departure control*", modelo binário de programação inteira, têm como restrições o número de partidas e chegadas possíveis num determinado intervalo de tempo, a capacidade de um sector no espaço aéreo e, por fim, o atraso máximo que é permitido.
2. A azul, tem-se um procedimento, que dependendo da confiança que se tem na assertividade das previsões climáticas, pode ser "executado" em intervalos entre 5 a 30 minutos. Com o objetivo final de assegurar que as aeronaves em fase de voo *en-route* evitam zonas onde a existência de condições atmosféricas adversas é uma realidade, o procedimento começa com a definição de regiões no espaço aéreo onde os níveis de perigosidade são não inferiores a 3. Após a demarcação das áreas que constituem um perigo, "*Trajectory Intersects Weather Check*" trata de determinar se as rotas dos voos intersectam as áreas anteriormente marcadas. Já em "*Deterministic Weather Rerouting*", para voos onde a rota precisa de ser redirecionada, é resolvido um problema linear baseado num algoritmo de caminho mais curto, onde numa fase inicial, são criados nós, à semelhança de um grafo (ver fig.3.13) e após solução do problema têm-se uma nova rota (fig.3.14).
3. Por fim, executado em intervalos de um minuto, o nível localizado mais a baixo na fig.3.12, serve para, em caso dos procedimentos acima mencionados falharem, isto é, impossibilidade

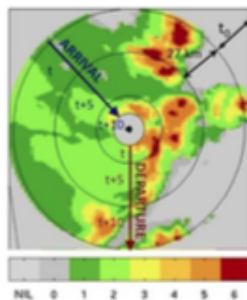


Figura 3.11: Exemplo dado pelos autores de condições previstas Vs condições reais. Retirado de [Pfeil and Balakrishnan, 2012]

de atrasar antes da partida e a impossibilidade de criação de um desvio na rota, pedir a realização de um *airborne holding*.

Ao testar a abordagem, os autores *Shon Grabbe and Banavar Sridhar (2009)* [Grabbe et al., 2009], realizam diversos testes entre eles, "*Sector Capacity versus Rerouting Look-ahead Distance*", "*Rerouting Frequency versus Rerouting Look-ahead Distance*". No primeiro teste foi possível observar que o numero de atrasos existentes pouco se altera até reduções de 30 pontos percentuais das capacidades dos sectores, sendo que o mesmo não se observa para casos onde um sector apenas possui 60 pontos percentuais da capacidade "normal". É também dedutível que até perdas de 50 pontos percentuais na capacidade dos sectores, existe um menor número de atrasos para *look-ahead* inferiores a 200nmi. Num segundo teste, 3.15b, é possível notar a rápida diminuição de atrasos com o aumento da frequência de *rerouting* e o seu rápido aumento com o diminuir do *look-ahead*.

3.3 Plataformas em uso

Neste capítulo, ao contrário da "Revisão Bibliográfica"(ver secção.3), são descritos, não abordagens, mas *softwares* que prestam auxílio valiosíssimo aos controladores aéreos, na medida em que conseguem obter informações bastante mais rápidas.

3.3.1 NAVCANatm

NAV Canada, empresa canadiana responsável pelo controlo do tráfego aéreo em espaço aéreo canadiano e, em espaço internacional, oferece ajuda a algumas rotas entre o continente americano e europeu assim como entre o continente americano e asiático [Airport-technology, 2017]. Segundo a empresa, *NAVCANatm* é uma linha de produtos criados com o propósito de auxiliar os controladores no controlo aéreo. Para isso, existem 7 módulos presentes na arquitetura do sistema, sendo que alguns estão descritos abaixo.

NAVCANstrips, módulo responsável pelo controlo de informação critica sobre os voos. Permite ainda aos controladores controlar as informações com um ecrã de toque, inutilizando os tradicionais papéis como forma de obterem, de forma organizada, as informações necessárias.

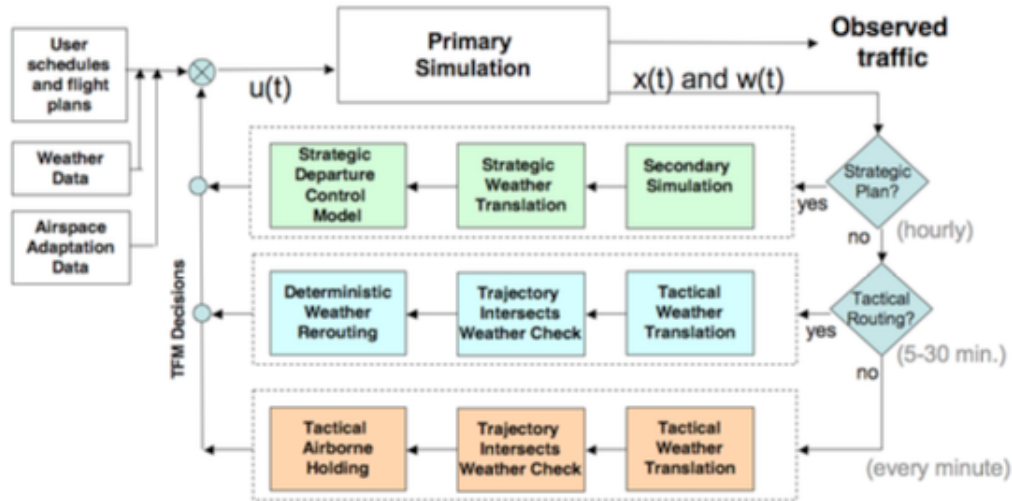


Figura 3.12: Arquitetura do modelo criado pelos autores do estudo. Retirado de [Grabbe et al., 2009]

NAVCANsitu, de acordo com [CANADA, 2017], permite de forma facilitada, ferramentas que ajudem os controladores a manterem as rotas dos aviões separadas.

NAVCANinfo é o módulo responsável pela demonstração de dados referentes ao aeroporto e às condições climáticas que se fazem sentir.

NAVCANatis é um sistema responsável pela transmissão de informação, como condições meteorológicas, configuração das pistas de aterragem e de partida, às aeronaves que se encontram prestes a aterrar ou a descolar.

Por fim, *NAVCANflow*, destaca-se dos restantes módulos por ser uma ferramenta cujo objetivo passa pelo planeamento de fluxo das aeronaves nos aeroportos.

3.3.2 STARS (Standard Terminal Automation Replacement System)

Segundo a empresa que o desenvolveu, este sistema providencia informação vital usada não só pela *FAA* como pelo departamento de defesa norte-americano. Com a junção de dados provenientes de mais de 255 radares e centenas de torres de controlo, *STARS* é capaz de conseguir localizar até 1350 aeronaves numa zona terminal. *STARS*, segundo a *FAA* [FAA, 2016b], é capaz de providenciar serviços como distanciamento e ordenamento de tráfego aéreo, meteorológicos e aplicação do procedimento de *radar vectoring* para aeronaves a descolar e a aterrar. Na fig. 3.16 é possível ver a aparência e a forma de como algumas das informações estão dispostas visualmente nos monitores de alta resolução usados pelos controladores aéreos.

3.3.3 VATSIM

Com o objetivo de se criar um modo de simulação bastante mais próxima da realidade, foi criada a organização *VATSIM* (Virtual Air Traffic Simulation Network). À semelhança do que

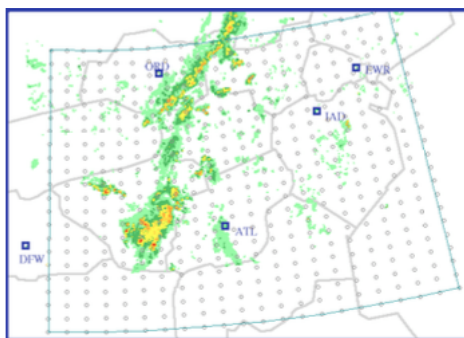


Figura 3.13: Criação de nós ao longo das regiões norte-americanas em exemplo dado pelos autores. Retirado de [Grabbe et al., 2009]

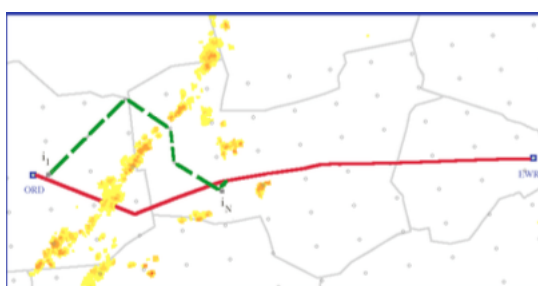
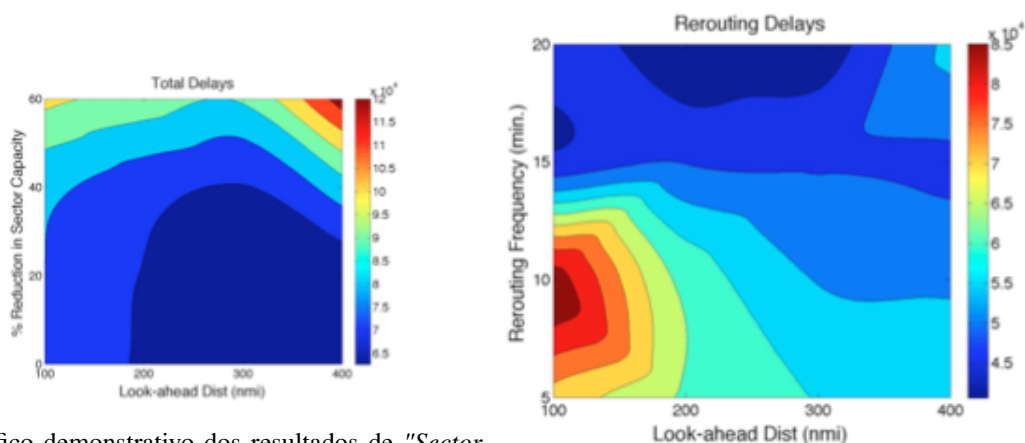


Figura 3.14: Desvio na Rota calculado através do problema linear em exemplo dado pelos autores. Retirado de [Grabbe et al., 2009]

acontece no mundo real, na plataforma, um controlador aéreo terá funções como providenciar ajuda aos pilotos, assegurar a separação das aeronaves querem as mesmas se encontrem em terra ou no ar, e manter um fluxo ordenado de tráfego [VATSIM.net, 2017]. Esta plataforma, sendo a mesma usada por pilotos e controladores, fornece a comunicação entre os mesmos por um serviço *VoIP* (*Voice over Internet Protocol*), contudo é necessário o uso de *software* alheio de forma a simularem um piloto ou um controlador. No caso do controlador aéreo, cujas algumas das suas funções são estudadas na atual dissertação, é necessário a utilização de aplicações que simulem radares. Sendo as aplicações grátis, os utilizadores podem optar por um variado número de opções, como *Virtual Radar Client*(fig 3.17a) que foi desenvolvido a pensar na facilidade de uso ao invés do realismo, ou *vERAM* que pretende implementar de forma muito semelhante, as funcionalidades que o sistema utilizado pela FAA, *En Route Automation Modernization* [FAA, 2015b], disponibiliza.

3.3.4 NextGen

Criado numa parceria entre a NASA (*National Aeronautics and Space Administration*) e a organização responsável pelo controlo do espaço aéreo norte-americano, FAA, *NextGen* (*Next Generation Air Transportation System*) permite uma melhor gestão do espaço aéreo. De forma a atingir objetivos como redução do congestionamento aéreo, melhorias no consumo de combustíveis e melhorar comunicação entre todos os envolvidos do espaço aéreo, foram criadas, ao todo,



(a) Gráfico demonstrativo dos resultados de "*Sector Capacity versus Rerouting Look-ahead Distance*"

(b) Gráfico demonstrativo dos resultados de "*Rerouting Frequency versus Rerouting Look-ahead Distance*"

Figura 3.15: Retirado de [Grabbe et al., 2009]

13 componentes, sendo que algumas das mais interessantes são [Houston, 2016a]:

- ***Automatic Dependent Surveillance-Broadcast***

Ao usar um sistema semelhante ao sistema de *GPS* em detrimento do sistema de radares, é possível, aos controladores aéreos terem informação cruciais sobre várias aeronaves, como velocidade e altura em tempo real.

- ***Low-Visibility Operations***

De forma a facilitar a tarefa de aterragem de uma aeronave em condições adversas à mesma, ajudando assim na diminuição do tráfego aéreo, *NextGen* providencia melhorias a certos sistemas em uso nas aeronaves de forma a permitir às mesmas fazerem uma aterragem automática.

- ***Airport improvement Program***

Componente responsável pelo estudo de como aumentar a capacidade de um aeroporto. Através do uso desta componente é possível a melhoria na eficiência das pistas de aterragem tendo em atenção, por exemplo, ao *Wake Recategorization*.

De facto o sistema veio de tal forma melhorar a forma como o espaço aéreo norte-americano é gerido, que para além de reduzir o congestionamento aéreo, melhorar o consumo de combustível, entre outros, é esperado que seja capaz de permitir poupar 106 mil milhões de dólares americanos à *FAA*.

3.3.5 SESAR

Com a premissa de que a tecnologia existente nos sistemas de gestão do tráfego aéreo tem-se mostrado insuficiente para colmatar as necessidades atuais do espaço aéreo, foi criado, numa

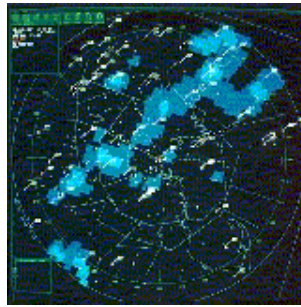
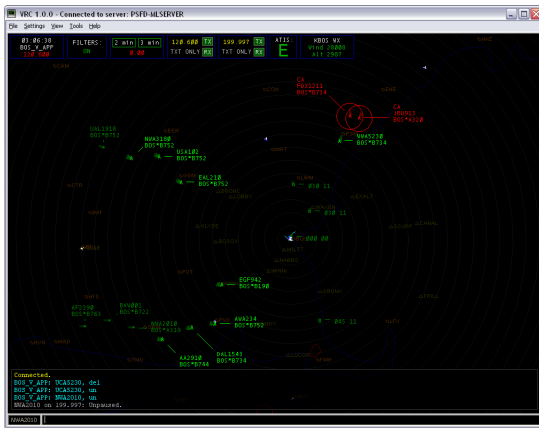
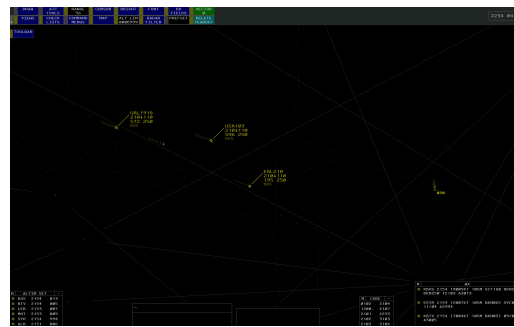


Figura 3.16: Figura ilustrativa da informação demonstrada pelo *STAR*. Retirado de [Raytheon, 2017]

união entre a União Europeia e *Eurocontrol*, um sistema inteligente de transporte aéreo europeu com o intuito de melhorar a gestão existente do espaço aéreo europeu [SESAR, 2016]. O intuito de aumentar a automatização de processos existentes no controlo de tráfego aéreo, a partilha de informação entre todos os envolvidos no uso dos transportes aéreos, assim como a possibilidade de o sistema se conseguir comunicar com outros sistemas são alguns dos pontos usados que *SESAR* (*Single European Sky ATM Research*) usa para se auto-retratar. Embora, no final de 2016, o programa tenha já permitido, entre outros, um aumento na capacidade dos aeroportos em 34 pontos percentuais reduzindo assim os atrasos existentes, é esperado para 2035 que o sistema seja capaz de, em média, diminuir o tempo de viagens em vinte minutos, assim como aumentar a segurança das aeronaves por um fator de três ou quatro.



(a) Visualização de um radar através de VRC



(b) Visualização de um radar através de vERAM

Figura 3.17: Retirado de [Metacraft, 2006] e [Metacraft, 2017] respetivamente.

Capítulo 4

Planeamento

O presente capítulo tem como propósito, numa primeira instância, a introdução à tecnologia já em uso no projeto de simulação sob o qual serão feitas as implementações propostas neste relatório e, numa segunda instância, será apresentada a abordagem idealizada para a realização do objetivo nuclear desta dissertação.

4.1 Tecnologias Utilizadas

Tendo em consideração que a plataforma que serve como base para o trabalho realizado nesta dissertação já foi utilizada no passado como em [Camara, 2013], [Sousa, 2010], [Câmara et al., 2014] e [Rodrigues et al., 2015], foi desejável a continuidade da utilização das várias tecnologias em uso. A linguagem C#, a plataforma de comunicação entre agentes *AgentService* (ver secção 4.1.2), o motor de simulação *Flight Simulator* (ver secção (4.1.1)) e a arquitetura de simulação da plataforma (ver secção 5.1) são algumas das características herdadas.

4.1.1 Microsoft Flight Simulator

Sendo considerado um dos mais famosos e antigos simuladores, *Microsoft Flight Simulator* foi desenvolvido pela *Microsoft Game Studios* em 1982 em forma de vídeo-jogo. Já depois do início do novo milénio, em 2006, e suportado pela introdução da, até à data, mais recente biblioteca relacionada com grafismo em ambiente *Windows*, *DirectX 10*, foi lançada a versão que é usada atualmente no projeto, *Microsoft Flight Simulator X*.

A possibilidade de adição de novas componentes ao jogo, sejam modelos de aeronaves ou controladores de tráfego aéreo, foram algumas das características usadas como forma de publicitar o simulador. Aliado ao facto do simulador já possuir um motor de jogo com componentes necessárias ao desenvolvimento do programa (e.g Físicas), o mesmo possui uma interface de comunicação (*API - Application Programming Interface*) bem documentada e capaz de alterar múltiplas variáveis da simulação.

Dada a qualidade do motor de simulação e da facilidade de comunicação oferecida pelo *SimConnect* fez deste simulador o mais adequado para utilização na plataforma.

4.1.1.1 *SimConnect*

O desenvolvimento desta interface, de autoria da gigante norte-americana *Microsoft*, tornou exequível a troca de variáveis da simulação entre o simulador e aplicações de terceiros.

Através de uma *API* onde a troca de informações rege-se por um modelo centrado em eventos, a interface *SimConnect* faz uso de um modelo *client-server* onde as respostas por parte do servidor são assíncronas [Microsoft, 2008]. Entre as múltiplas funcionalidades que esta interface permite desenvolver, podendo-o fazer num variado leque de linguagem de programação como C, C++ ou .NET, encontra-se a possibilidade de controlar as mais diversas variáveis dos vários objetos que constam no *Microsoft Flight Simulator*.

4.1.2 *AgentService*

Sendo o projeto a desenvolver, um sistema multi-agente era necessário a implementação ou uso de uma camada de comunicação capaz de aguentar o fluxo de mensagens trocadas na plataforma.

Com o objetivo de facilitar a criação de sistemas multi-agentes, providenciado para tal uma plataforma flexível, robusta e capaz de ser usada em vários sistemas operativos, os autores de [Vecchiola et al., 2008] desenvolveram a plataforma *AgentService*. Baseada em tecnologia .NET e construída de acordo com as recomendações propostas pela *FIPA (Foundation for Intelligent and Physical Agent)*,

Assim, o *AgentService*, capaz de estabelecer conexão com sistemas externos ao próprio e de já ser uma plataforma em que a qualidade se mostrava assegurada, foi a solução escolhida para ser responsável pela comunicação entre os vários agentes da plataforma.

4.2 Abordagem ao problema

O problema basilar desta dissertação centra-se numa melhor gestão do espaço aéreo nas proximidades dos aeroportos de forma a reduzir o congestionamento na área. Após uma leitura detalhada de várias abordagens descritas no Capítulo 3, foram retiradas algumas ideias-chaves capazes de fazer diminuir o congestionamento aéreo. Esta secção tem então como objetivo o relatar das soluções provisórias previstas, sendo que se dividiu o problema em duas partes.

4.2.1 Diminuição do congestionamento do tráfego aéreo

Nesta subsecção são expostas as soluções pensadas para que, na eventualidade de um possível congestionamento, o sistema seja capaz de evitar o seu acontecimento ou, no pior das hipóteses, não o piorar. De forma a que se consiga evitar um congestionamento de tráfego aéreo de acontecer ou de tomar dimensões ainda mais preocupantes é pretendida a implementação **de comunicação os vários controladores ATC** e ainda a criação de um módulo capaz **gerar novas rotas**. Através

da comunicação entre controladores *ATC* seria possível aos vários aeroportos de origem atrasarem o horário de partida de voos cujo destino, por exemplo, informou a existência de condições adversas que impeçam a aterragem das aeronaves no horário previsto da chegada do voo. Contudo a comunicação apenas facilita os casos passíveis de prever com antecedência. Em casos em que a aeronave já se encontra a caminho do destino é necessária a criação de rotas, mais longas ou mais curtas, de forma a que se consiga atrasar/adiantar o horário de chegada.

4.2.2 Lidar com o congestionamento aéreo

Como forma de lidar com o congestionamento aéreo existente no espaço aéreo dos aeroportos pretende-se, para além do **uso do módulo de criação de rotas, melhorar a eficiência das pistas de aterragem** assim como **melhorar a forma como serão utilizados os locais usados para a realização das manobras de espera**. Para que se diminua o número de pedidos de permissão para aterrar por intervalo de tempo, pretende-se usar o módulo para criação de rotas, como forma de gerar uma rota maior ou menor, dependendo dos casos, para que as aeronaves que estejam a aproximar-se do aeroporto consigam chegar de uma forma faseada. Atualmente, no sistema já existente, apenas pode existir uma aeronave por local de espera, o que implica problemas de gestão de espaço aéreo caso haja um congestionamento na área. A solução prevista para este problema é então a estratificação do espaço aéreo através da introdução da componente altura. Com a implementação desta solução, é esperado que seja então possível, para múltiplas aeronaves, aguardar permissão para aterrar no mesmo local predefinido para o efeito, estando separadas de acordo com as regras por uma distância não inferior a 1000ft (ver Fig. 4.1).

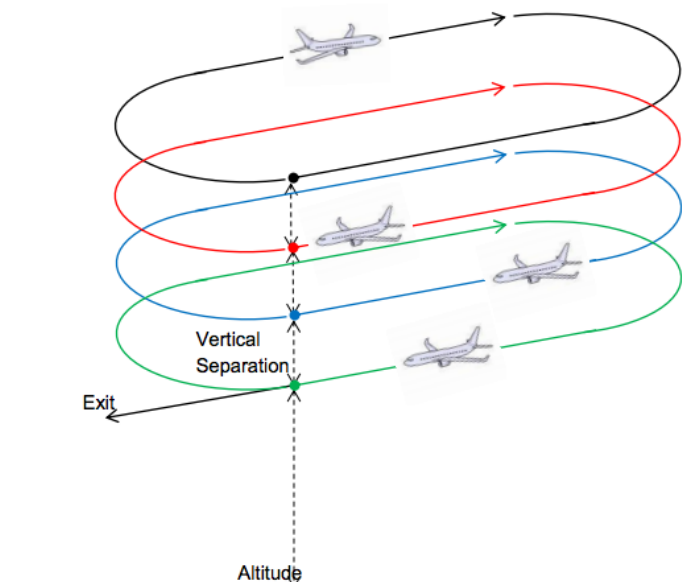


Figura 4.1: Exemplo da estratificação usada na gestão de aeronaves em espera [IVAO, 2016].

Dado o fenómeno de *Wake Turbulence*, a *FAA* pré-estabeleceu um conjunto de regras de forma a garantir a segurança das várias aeronaves que pretendem aterrar num determinado aeroporto. A

Planeamento

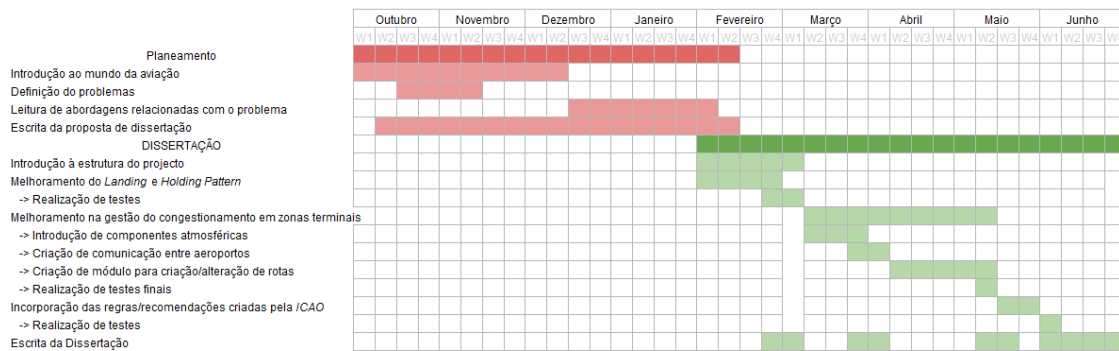


Figura 4.2: Planeamento das tarefas a realizar dispostas num gráfico Gantt

regra especificamente aplicada ao trajeto de aproximação às pistas tem como objetivo a separação temporal entre aterragem, sendo que uma pista pode ter de ficar inativa entre dois minutos até quatro minutos. Tendo esta regra em conta, é pretendida a criação da funcionalidade de sequenciar as várias aeronaves de forma a aumentar a taxa de aterragem de um aeroporto.

4.3 Planeamento e Análise de risco

Finda a definição do problema e dos objetivos para a presente dissertação foi criado um planeamento onde, através da Fig. 4.2, é possível verificar que se separou o futuro trabalho em dois *sprints*. O primeiro, a ocorrer durante o mês de fevereiro, tem como objetivo a realização do melhoramento da gestão da fila de espera. Com a manobra de espera já implementada, serve também o presente *sprint* para ajudar a ambientar-se à forma como o projeto se encontra estruturado. Numa segunda fase tenciona-se a criar o módulo responsável pela comunicação entre aeroportos através de um sistema central, que como descrito no capítulo 4, servirá para diminuir o congestionamento presente nas proximidades dos aeroportos. Por fim, termina-se com a implementação das regras/recomendações impostas pela ICAO. É ainda de realçar que a escrita da dissertação ocorrerá aquando sempre na fase final da programação e fase de testes.

Nesta secção são enumerados os vários riscos que estão presentes durante o decorrer da presente dissertação assim como as respetivas estratégias de forma a diminuir/suavizar os seus efeitos. Com base no atual cenário dividiu-se os vários riscos existentes em duas categorias, riscos tecnológicos e humanos. Numa primeira instância tem-se a categoria onde se encontram agrupados os riscos relacionados com as plataformas/serviços usados na simulação. Ao iniciar a presente dissertação e tendo uma ideia de como serão os testes, uma das maiores preocupações reside na capacidade do serviço optado para a troca de mensagens entre agentes dado que é estimado que para além dos vários agentes *ATC* e *Control Panel* estejam acima de uma centena de aeronaves a requisitarem a plataforma de comunicação *AgentService*. De forma a diminuir a sobrecarga do serviço tenciona-se estender o uso do algoritmo de compressão de mensagens (visto em [Camara, 2013]), a várias outras mensagens onde o conteúdo seja de maior dimensão.

Tabela 4.1: Tabela dos vários riscos presentes no desenvolvimento da presente dissertação.

Tipo de risco	Risco	Probabilidade
Tecnológico	Sobrecarga do <i>AgentService</i>	Médio
Tecnológico	Dimensão da Simulação do <i>Flight Simulator</i>	Médio
Tecnológico	Informação errada em ficheiros XML	Alto
Humano	Atrasos no desenvolvimento	Médio
Humano	Implementação de funcionalidades em que são usadas informações erradas	Baixo
Humano	Desfasamento entre o tempo estimado e o tempo realmente necessário para o desenvolvimento de uma tarefa	Alto
Humano	Desfasamento entre a dificuldade estimada e a realmente sentida na implementação de funcionalidades	Médio

Se mesmo após o primeiro passo resolutivo, o serviço se mostrar impotente, será necessário aumentar o número de *AgentService* assim como a transição dos agentes entre os mesmos. Ainda com base nos testes e admitindo a necessidade de criação de aeronaves em diferentes aeroportos com distâncias consideráveis entre si, teme-se que o motor de jogo possua um território de simulação reduzido quando comparado com o necessário para a simulação. Caso de facto se verifique o problema, terá de lançar várias instâncias do motor de jogo, sendo que cada ficaria encarregue de monitorizar e simular em partes distintas do território. A solução passa então pela adaptação da solução vista em [Rodrigues et al., 2015] às restantes componentes da plataforma.

Sendo os dados contidos nos vários formatos *XML* obtidos através de ficheiros com origem num motor de jogo lançado em 2006, incorre-se no risco de existirem dados, como número de pistas ou aeroportos existentes, que se encontrem desatualizados. Não havendo a possibilidade de aceder a uma base de dados atualizada e oficial, a única solução prevista assenta na comparação manual dos dados conforme se vai trabalhando com os diversos aeroportos.

Relativamente aos riscos com origem no autor do presente trabalho, potenciais atrasos no desenvolvimento e implementação errónea de funcionalidades são os erros cuja gravidade se mostra mais preocupante. Reconhecendo a inexperiência em estimativas de durações de implementação por parte do autor existe a possibilidade de se incorrer em atrasos no desenvolvimento do projeto quando comparado com o plano. A aliar-se a estes dois riscos, existe o perigo de uma funcionalidade poder mostrar-se mais complexa do que se estava à espera seja pela dificuldade ou pela necessidade de implementação de funções que já seria esperado estarem implementadas. De forma a reduzir as potenciais consequências que advêm da ocorrência deste problema, será necessário a discussão com o orientador e co-orientador de forma a avaliar se as várias tarefas a realizar trazem de facto algum valor para a plataforma/simulação. Como forma de solucionar o segundo risco desta categoria, para além de uma atenção redobrada na leitura das informações é necessário que se escolha com um critério adequado as fontes das mesmas. Relativamente ao terceiro risco, a

Planeamento

solução passaria pela adaptação do plano original, onde poderia ser necessário a remoção de um objetivo de menor valor em detrimento das tarefas de maior grandeza.

Capítulo 5

Implementação

No presente capítulo serão apresentados as várias funcionalidades que foram sendo construídas ao longo deste projeto.

Começar-se-á pela exposição das alterações realizadas na arquitetura da plataforma, avançando-se para a reconstrução do módulo responsável pela gestão de filas de aterragem, conforme as normas da *FAA*, seguindo-se uma descrição detalhada do novo tipo de *ATC* adicionado ao projeto e terminar-se-á na explicação detalhada das alterações dos dialetos e visuais da plataforma.

5.1 Arquitetura do sistema

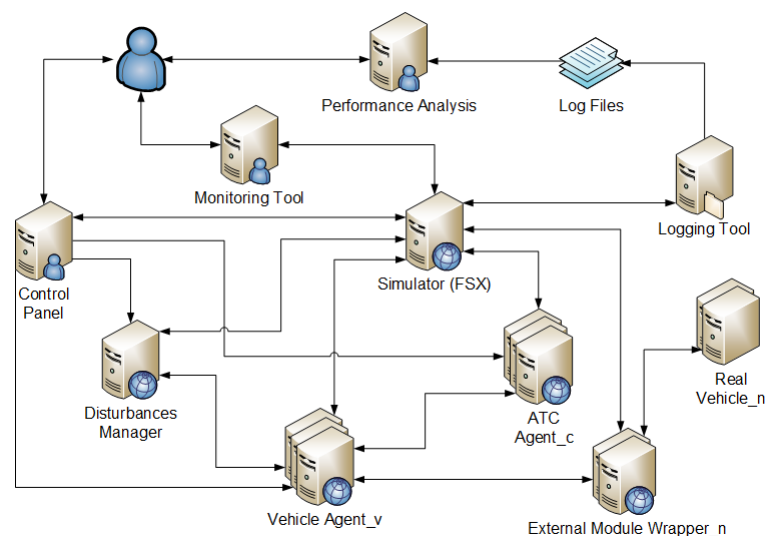


Figura 5.1: Arquitetura de componentes que compõem a plataforma.

Visto que a separação entre os vários agentes que integram a plataforma é necessário, a arquitetura composta por dez componentes com funções distintas e demonstrada na figura 5.1 foi projetada em [Silva, 2011]. Nesta secção serão descritas as várias componentes para uma melhor compreensão da forma como a plataforma opera.

Implementação

Sendo a componente mais nuclear da plataforma já que se encontra interligada, através do *SimConnect*, com todos os restantes elementos, o *Simulator* serve de motor de jogo capaz de simular os vários eventos respeitando as mais variadas leis físicas. Sendo requisitado ou ativando-se por eventos, o módulo transmite informação necessária para que os vários agentes consigam tomar as decisões necessárias.

Uma das componentes que está constantemente a receber informação por parte do simulador é denominada de *ATC Agent*. Tendo como objetivo simular um controlador aéreo este elemento é capaz de fazer não só uma gestão ativa das infraestruturas dos vários aeroportos mapeados como também uma gestão organizada do espaço aéreo e das aeronaves que circulam entre ele.

A componente *Vehicle Agent* é responsável pelo controlo de uma aeronave, sendo que para isso, permite, entre outros, alterar múltiplas variáveis como direção, altitude e velocidade. Ao adicionar ao controlo da aeronave que representa, uma instância deste elemento da plataforma possibilita a monitorização através do *google maps* e a alteração do ponto de destino da aeronave.

Sendo capaz de comunicar com os últimas duas componentes, *Control Panel* dá a possibilidade ao utilizador de alterar a simulação em tempo de execução. Exemplos deste controlo são a alteração e o lançamento das várias aeronaves que constituem as várias equipas (companhias aéreas) e a possibilidade de lançar as várias missões que existem e que representam tráfego real.

5.1.1 Agente ATC

Mostrando-se como a componente que sofreu mais alterações, visto que o objetivo basilar desta dissertação incidia sobre ela, faz sentido descrever as várias modificações na sua arquitetura interna. Embora a divisão em três categorias distintas tenham sido mantidas, novos módulos foram adicionados como ocorreu na categoria *ATC*, ou ocorrem alterações dentro dos módulos já existentes. De seguida são descritas as várias diferenças desta arquitetura quando comparada com a sua original [Camara, 2013] assim como são resumidas os vários módulos já existentes e que ficaram inalterados.

Sendo a única categoria que se manteve inalterada a nível da sua funcionalidade, a camada de conexão faz uso dos seus dois sub-módulos como forma de cumprir com o seu propósito. Assim, e para obter informações provenientes do simulador, é usado o *simConnect* enquanto que para comunicar os os demais agentes em execução faz-se uso da plataforma *agentService*.

Apesar de não ter sofrido modificações tão drásticas como a categoria *ATC*, a categoria responsável pela representação do conhecimento estático e obtido em tempo de execução possui uma nova fonte de informação, estando a mesma representada por *Missões* na figura 5.2. O primeiro sub-módulo presente em *XML* representa toda a informação que está armazenada no ficheiro de configuração do cenário de simulação. Até à presente dissertação esta informação embora fidedigna mostrava-se incompleta e, como tal, foi necessária a adição de nova informação como a descrição dos vários nós do espaço aéreo, as várias rotas usadas pelas aeronaves nos vários espaços aéreos, entre outros. Contrariamente ao sub-módulo *Conhecimento* que se mostra constante ao longo da simulação, o sub-módulo *Radar* tem como principal objetivo ser capaz de representar as mais variadas informações sobre as aeronaves que se encontram ou que se encontraram sob

Implementação

o controlo do agente *ATC*. Desta forma, conforme os dados das múltiplas aeronaves vão sendo recebidos através da camada de conexão, este sub-módulo adiciona/modifica as informações que o agente possui acerca das mesmas. Em relação à nova fonte de informação, *Missões*, e tal como a sua denominação sugere, representa as várias missões (nomenclatura dada aos voos das várias companhias aéreas) existentes. Os vários sub-módulos presentes em *XML* representam informação imutável e fidedigna e como tal não existe a necessidade das informações serem alteradas após o sua leitura por parte do agente *ATC*, sendo as mesmas então carregas apenas uma vez no início da simulação.

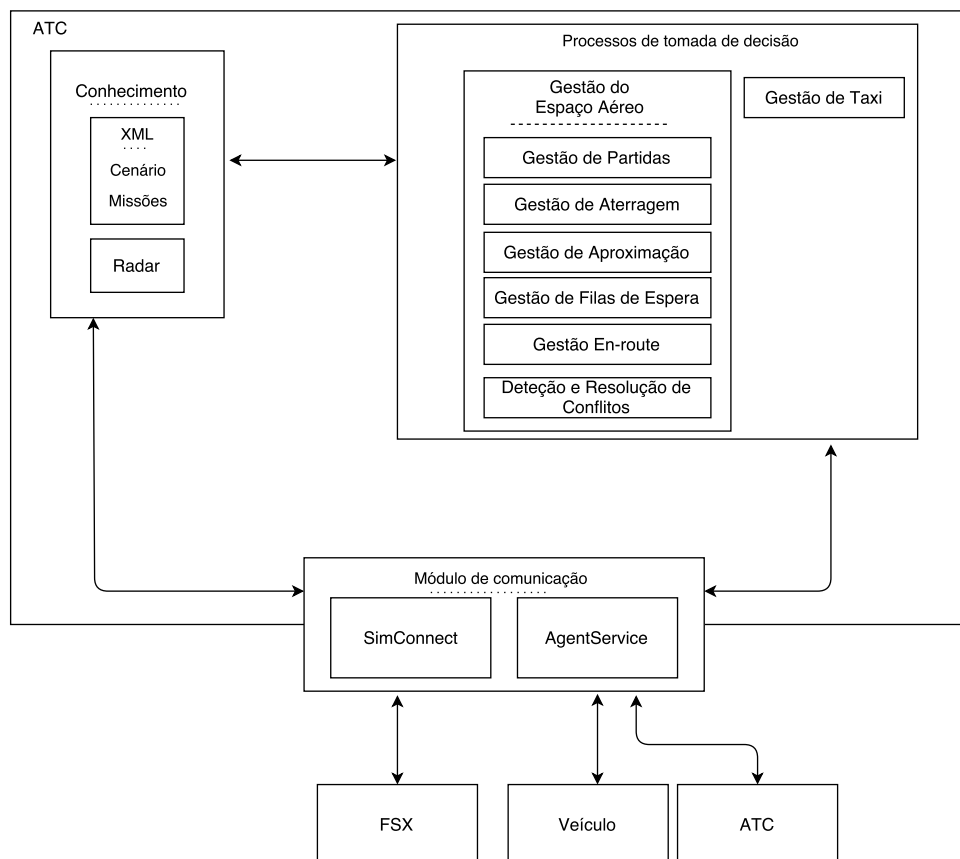


Figura 5.2: Arquitetura dos módulos que constituem a componente do ATC.

Dada a separação do, até então, único tipo de controlador aéreo que existia e do melhoramento implementado na forma como é gerido os diferentes espaços aéreos, tornou-se indispensável a alteração da arquitetura da categoria *ATC*. Em [Camara, 2013] existiam os módulos responsáveis pela separação de aeronaves que se encontrassem em voo (*Deteção e resolução de conflitos*), pela aterragem e pela partidas das várias aeronaves (*Aterragem e "Decolagem"* respetivamente) e, por fim, um sub-módulo responsável pela gestão das aeronaves que se encontrassem em *taxi*. No entanto, dado a criação de um novo controlador aéreo responsável pela gestão de aeronaves fora das zonas monitorizadas pelos aeroportos aliado à implementação de dois novos tipos de gestão, foi imprescindível a criação de dois sub-módulos, *Gestão de filas* e *Gestão de aeronaves en-route*. De maneira de representar o tipo de gestão a ser utilizada pelos dois tipos de controladores

temos o sub-módulo de *gestão de filas*. Fazendo uso dos vários tipos de informação, estática ou dinâmica, tenta melhorar a eficiência de um aeroporto sendo que para isso tenta reduzir a quantidade de atrasos assim como a duração dos mesmos. É de salientar que uma das principais vantagens ganhas com a implementação deste sub-módulo foi a aproximação da logística retratada na simulação à realidade. Com o objetivo de conseguir controlar as várias aeronaves que transitam entre espaços aéreos de aeroportos, foi necessária a implementação de *Gestão de aeronaves en-route*. Este módulo faz uso da informação do espaço aéreo assim como dos vários voos existentes de modo a ser capaz de controlar e diminuir potenciais congestionamentos que possam surgir na simulação.

No decorrer da dissertação foi introduzido um novo módulo de gestão de filas de aterragem com recurso a manobras de espera com o objetivo de substituir o sistema já existente e de aproximar a plataforma ao mundo aeronáutico. Este módulo alberga uma nova forma de aproximação às pistas no momento de aterragem, um sistema de gestão de filas de espera renovado e realista e ainda um novo algoritmo usado para escolha de pistas a serem usadas.

5.2 Rotas de aterragem e de partida das Aeronaves

Como forma de evitar a utilização de rotas fictícias para a fase inicial e final de um voo foi impreterível a obtenção das rotas praticadas na realidade e o respetivo processamento para a modificação do ficheiro XML que contém as informações detalhas das infraestruturas dos aeroportos.

Acedeu-se então aos mapas da *FAA* (ver Fig. 2.12) e converteu-se a informação presente para um ficheiro *CSV- Comma-Separated Values*, sendo que foram retirados dados como nome e se o próprio ponto é utilizado como zona para a realização de manobras de espera. Não contendo a localização (latitude, longitude) foi necessário recorrer a outro serviço *web* [FAA, 2017] como forma de obter a localização geográfica. Através da Tabela 5.1 é passível de ver os vários atributos que se obtiveram para cada aproximação, sendo que cada aproximação é composta por pontos de aproximação à pista e por pontos caso a aeronave falhe a aterragem (*Missed Approach*). De forma a definir a manobra de espera foi necessário a inclusão de dados como o a direção da manobra (*Holding Degrees*) e se o tipo de manobra a ser realizado (*Orientation*), que pode tomar o valor "R" caso seja standard e "L" caso contrário.

Tabela 5.1: Exemplo de aproximações às pistas 6L e 6R do aeroporto de Cleveland, OHIO.

Runways	Waypoints	Lat	Long	Altitude	Holding?	Missed App	Holding Degrees	Orientation
6L	DUPAY	41° 16' 11.3800"N	82° 04' 41.4400"W	3000	TRUE	FALSE	57	R
	SASCO	41° 20' 04.0300"N	81° 58' 35.8700"W	2800	FALSE	FALSE		
	ACAPO	41° 26' 33.6800"N	81° 48' 20.8600"W	3000	FALSE	TRUE		
	PASLE	41° 34' 34.9500"N	81° 49' 36.3900"W	3000	TRUE	TRUE	54	L
6R	SERLE	41° 16' 15.3500"N	82° 04' 10.2100"W	3000	TRUE	FALSE	57	R
	TUBOE	41° 20' 08.2500"N	81° 58' 04.0400"W	2800	FALSE	FALSE		
	MOMBE	41° 27' 17.1200"N	81° 46' 46.6200"W	3000	FALSE	TRUE		
	LEBRN	41° 27' 08.0100"N	81° 38' 36.2200"W	3000	TRUE	TRUE	278	R

Após a conversão dos dados de aproximação de todas as pistas de um aeroporto, o ficheiro serve como argumento de entrada para um *script* escrito em linguagem Python cuja finalidade é a

Implementação

transcrição da informação obtida e representada no ficheiro acima para formato XML. O resultado, que pode ser observado na Listagem 5.1, é depois adicionada ao ficheiro XML de forma a que a informação possa ser carregada e usada pela plataforma.

```
1
2 <approachRoutes>
3   <approachToRunway runway="6L">
4     <approach>
5       <approachWaypoints>
6         <approachWaypoint name="DUPAY" hasHolding="true" holdingDegrees="57"
7           holdingDir="R">
8           <coordinates>
9             <latitude>41° 16' 11.3800''N</latitude>
10            <longitude>82° 04' 41.4400''W</longitude>
11            <altitude lengthUnit="Foot" measured="amsl">3000</altitude>
12          </coordinates>
13        </approachWaypoint>
14        <approachWaypoint name="SASCO" hasHolding="false">..</approachWaypoint>
15      </approachWaypoints>
16      <missedApproach>
17        <approachWaypoint name="ACAPO" hasHolding="false">
18          <coordinates>
19            <latitude>41° 26' 33.6800''N</latitude>
20            <longitude>81° 48' 20.8600''W</longitude>
21            <altitude lengthUnit="Foot" measured="amsl">3000</altitude>
22          </coordinates>
23        </approachWaypoint>
24        <approachWaypoint name="PASLE" hasHolding="true" holdingDegrees="54"
25          holdingDir="L">...</approachWaypoint>
26      </missedApproach>
27    </approach>
28  </approachToRunway>
29  <approachToRunway runway="6R">...</approachToRunway>
30</approachRoutes>
```

Listagem 5.1: Exemplo das *tags* resultantes a adicionar ao aeroporto de Cleveland, OHIO.

Após a finalização das rotas de aproximação e, como forma de conseguir reproduzir também as rotas de partida de forma realista, foi necessário o uso de um procedimento semelhante ao usado para a representação das rotas de aproximação. As diferenças mais notáveis, entre as rotas de aproximação e de partida, é a partilha de rotas entre várias pistas e a falta de locais para a execução de manobras de espera nas rotas. Ao transcrever a informação obtida através das cartas oficiais, é-se obtida uma tabela semelhante à representada em 5.2.

Recorrendo novamente ao *script* em *Python* foi possível a conversão dos dados obtidos para formato XML, cujo formato está visível na figura 5.2.

Implementação

Tabela 5.2: Exemplo de rotas de partida no aeroporto de Cleveland, OHIO.

Departure	Lat	Long	Altitude	Name
6L/6R/10/24I/24R/28	41° 18' 40.58"N	82° 39' 35.35"W	3000	HUDDZ
	41° 14' 42.18"N	82° 59' 19.8"W	3000	AMRST
	42° 26' 9.24"N	84° 30' 35.7"W	3000	ALPHE
6L/6R/10/24I/24R/28	41° 21' 29.03"N	82° 9' 43.09"W	5000	DRYER
	41° 18' 40.58"N	82° 39' 35.35"W	5000	HUDDZ
	41° 14' 42.18"N	82° 59' 19.8"W	5000	AMRST
	41° 27' 5.4"N	83° 38' 19.2"W	5000	WATERVILLE

```

1 <departures>
2   <departure>
3     <fromRunway>
4       <lrway idr="6L" />
5       ...
6     </fromRunway>
7     <departureSequence>
8       <sequenceAlternative>
9         <altitude lengthUnit="Foot" measured="amsl">5000</altitude>
10      </sequenceAlternative>
11      <sequenceAlternative>
12        <Fixpoint name="AMRST">
13          <coordinates>
14            <latitude>41° 14' 42.18''N</latitude>
15            <longitude>82° 59' 19.8''W</longitude>
16            <altitude lengthUnit="Foot" measured="amsl">3000</altitude>
17          </coordinates>
18        </Fixpoint>
19        <Fixpoint name="ALPHE">...</Fixpoint>
20      </sequenceAlternative>
21    </departureSequence>
22  </departure>
23  <departure>...</departure>
24 </departures>

```

Listagem 5.2: Exemplo de rotas de partida no aeroporto de Cleveland, OHIO em formato XML.

A necessidade de criar estes mapeamentos mostra-se crucial não só por conseguir reproduzir as rotas usadas na realidade, como serve de base para outro procedimento cujo objetivo é a determinação de grupos de pistas que podem ser usados simultaneamente.

Os dados resultantes dos vários procedimentos expostos nesta secção têm como destino o ficheiro de configuração dos aeroportos carregados pelos agentes *ATC*.

No entanto, devido à distinção criada entre controladores de tráfego aéreo, ver secção 5.4, apenas os controladores de espaço aéreo responsáveis pela gestão do espaço aéreo de um aeroporto farão uso destas informações. Em jeito de conclusão, e tendo em consideração variáveis como proximidade existente entre aeroportos, fluxo de passageiros e a complexidade de voos existentes

nos aeroportos, foram mapeados quatro aeroportos de *Ohio*, *KCLE* (*Cleveland-Hopkins International Airport*), *KCMH* - (*Port Columbus International Airport*), *KDAY* - (*James M Cox Dayton International Airport*) e, por fim, *KCAK* - (*Akron-Canton Regional Airport*).

5.3 Manobras de Espera

Para além do melhoramento na forma como as aeronaves se deslocam dentro do espaço aéreo controlado pelos controladores dos aeroportos, foi necessária a implementação de um sistema capaz de organizar de forma eficiente as aeronaves, após as mesmas solicitarem permissão para aterrar. Tal como foi abordado na secção 5.2, tanto o início como o fim das rotas de aproximação podem ser usadas como locais para a realização das manobras de espera. Assim estes pontos foram usados como forma de auxílio aos controladores com o objetivo de manter o espaço aéreo organizado. Durante a criação do presente módulo foi necessária a implementação de várias medidas como forma de ajustar o presente módulo à realidade, sendo as mesmas analisadas de seguida.

5.3.1 Escolha de pistas a serem utilizadas

Anteriormente à realização deste módulo, todas as pistas se encontravam disponíveis para serem usadas por uma aeronave que requisitasse permissão para aterrar sendo que a única exceção era quando a pista se encontrava já em uso. Outra característica que desvalorizava o módulo era o facto de o mesmo não ter em consideração o tempo que uma aeronave iria esperar para aterrar já que atribuía sempre a pista mais próxima da aeronave. Para além de não tentar minimizar o período temporal no qual a aeronave aguardava por permissão para dar início à aproximação final, o procedimento anterior acarretava graves problemas de segurança para as aeronaves, sendo que as distâncias mínimas entre aeronaves não eram respeitadas assim como poderia levar à utilização de pistas de forma simultânea sendo que possíveis colisões não eram tidas em consideração. De modo a encontrar uma forma de salvaguardar a segurança das aeronaves em fases iniciais, em caso de partidas, e em fases terminais, caso de chegadas, foi necessária a implementação de uma forma eficaz de determinar pistas que possam ser usadas simultaneamente ou intercaladamente. Visto que a procura por uma fonte fiável de grupos de pistas que eram usados nos vários aeroportos usados mostrou-se ineficaz, a solução passou pela criação de um algoritmo capaz de, tendo em consideração a configuração realista das várias pistas e a distância entre elas, gerar grupos plausíveis de pistas que tanto pudessem ser usadas de forma simultânea como alternada.

Como se pode ver em 2.2.1 e não obstante ao facto de uma pista apenas poder ser utilizada por uma única aeronave simultaneamente, uma pista pode ser usada em ambas as suas direções. De forma a atribuir a direção a ser usada num determinado espaço temporal, recorre-se a fatores como a direção preferencial do tráfego aéreo, condições atmosféricas, entre outros. Com o intuito de determinar os vários grupos possíveis, o algoritmo considera que cada pista existente no aeroporto, como sendo quatro pistas diferentes (pistas lógicas), em virtude da combinação da direção operacional da pista e das diferentes manobras que podem ser utilizadas (partida ou descolagem).

Implementação

Após a determinação e a construção em memória das várias pistas lógicas, o algoritmo criado possui três passos distintos sendo que o primeiro passa pela criação de polígonos, numa segunda instância o algoritmo testa a colisão dos mesmos polígonos entre si de forma a conseguir encontrar grupos de polígonos que não se cruzem e, por último, tenta criar os grupos plausíveis. Como foi visto na secção 5.2, para cada pista lógica têm-se uma rota de aproximação assim como uma rota de partida a serem usadas pelas aeronaves. Assim, para cada pista lógica e para as diferentes operações passíveis de serem usadas, o algoritmo cria polígonos com uma largura mínima definida pela *FAA*. Ao contrário das rotas de aproximação, cujos polígono têm início no primeiro ponto de aproximação e terminam no último ponto do *missed approach* (ver Fig. 5.3), os polígonos de partida para além de possuir a dimensão da largura mínima imposta pela *FAA* para este tipo de operações, é criado um semi-cone de 15° que se estende por um quilómetro. Dado que uma aeronave pode sair para qualquer um dos lados, este semi-cone é criado, no momento do teste de colisão, tendo em consideração a localização da pista que se encontra a ser comparada.



Figura 5.3: Rota de aproximação, base para criação do polígono, da pista 6L de Cleaveland, OHIO à esquerda, e polígono consequente à direita.

Como método de obtenção dos grupos de pistas é criado um grafo onde os nós existentes representam os tuplo de pistas lógicas com operações e as adjacências entre dois nós representam a possibilidade de serem usadas simultaneamente (Tuplo<PistaLógica, Operação>). Com o decorrer das comparações entre os vários polígonos, o grafo vai ganhando adjacências como mostra a Fig. 5.5.

Finda a comparação entre os vários polígonos é necessário a conjugação das várias possibilidades em grupos (ver Algoritmo 1). Sendo o principal objetivo do algoritmo a criação de grupos a englobar o maior número de pistas possível era necessário uma processamento da informação do grafo devido a possibilidade dos vários nós conectados a um determinado nó não estarem conectados entre si. Assim, de forma a gerar grupos, o algoritmo retira um nó do grafo marcando-o como

Implementação

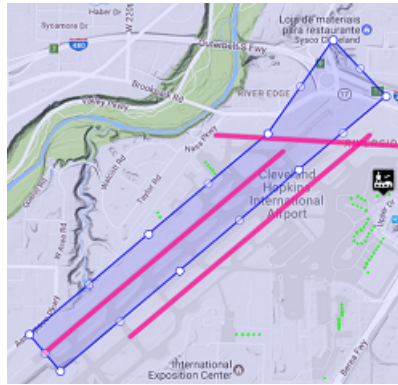


Figura 5.4: Polígono resultante para a pista 6L de Cleveland, Ohio para a operação de partida.

"pai" de um grupo. Após a extração é necessário adicionar os nós adjacentes, sendo que para isso, é necessário verificar se o nó a adicionar possui ligações a todos os nós já pertencentes ao grupo. Em caso negativo, é criado um segundo grupo contendo os mesmos nós pertencentes ao primeiro com a única exceção dos nós que não possuem adjacências ao nó a ser adicionado.

Algorithm 1 Algoritmo criado para obtenção de grupos de pistas

```
1: graph is a global variable
2: procedure CREATEGROUPS
3:   groups = list of lists of nodes
4:
5:   for all (string, airportOperation) rnw in graph.nodes do
6:     currentGroup = list of nodes
7:     possibleNewGroup = list of nodes
8:     runwaysUsed = list of strings
9:     edges = graph[rnw].getAdj()
10:    possibleNewAdj = stack of Edges
11:    allEdgesUsed = false
12:
13:    currentGroup.Add(rnw)
14:    runwaysUsed.Add(nos[rnw].runwayID)
15:
```

▷ Continua na página seguinte.

Após a criação de grupos de pistas que podem ser usadas coincidentemente, e como forma de aumentar o fluxo de operações realizadas, tenta-se adicionar pistas que possam ser usadas de forma alternada com as pistas já presentes no grupo. Sendo o objetivo destas pistas poderem ser usadas enquanto as pistas "principais" por questões de segurança não podem ser usadas temporariamente, as mesmas podem ser usadas alternadamente caso a rota das operações não coincidirem com nenhuma das "principais". Ao invés de usar polígonos com as distâncias recomendadas pela FAA, usam-se polígonos que apenas contêm as pistas e as respectivas rotas de aproximação, em caso da operação ser aterragem, ou de partida, caso contrário. Com base no conhecimento da demanda de operações que o aeroporto vai ter no próximo intervalo de tempo, a plataforma altera

Algorithm 1 Algoritmo criado para obtenção de grupos de pistas (continuação)

```

16:   while !allEdgesUsed do
17:     while edges.Count != 0 do
18:       rnway = edges.Pop()
19:       canBeAdded = false
20:       for i = 0; i < currentGroup.Count; i++ do
21:         if node with runwayID and operation airop has an edge to  

rnway.destination then
22:           canBeAdded = true
23:           break
24:         if canBeAdded then
25:           add rnway.destination runwayID to currentGroup
26:           add rnway.destination runwayID to runwaysUsed
27:           for all Edge ed in graph[(rnway.final.runwayID, rnway.final.airop)].getAdj()
do
28:             edges.Push(ed)
29:           if currentGroup.Count > 1 then
30:             groups.Add(currentGroup)
31:           else
32:             runwayNotUsedInGroup = true
33:             for all List of nodes g in groups do
34:               if g.Contains(currentGroup[0]) then
35:                 runwayNotUsedInGroup = false
36:                 break
37:             if runwayNotUsedInGroup then
38:               groups.Add(currentGroup)
39:
40:           remove all edges in graph between nodes in currentGroup
41:
42:           possibleNewGroup = list of nodes
43:           possibleNewGroup.Add(graph[rnw])
44:           possibleNewAdj = graph[rnw].getAdj()
45:           for all Edge ed in graph[rnw].getAdj() do
46:             if !runwaysUsed.Contains(ed.final.runwayID) then
47:               possibleNewGroup.Add(ed.final)
48:               runwaysUsed.Add(ed.final.runwayID)
49:             for all Edge edp in graph[(ed.final.runwayID, ed.final.airop)].getAdj()
do
50:               possibleNewAdj.Push(edp)
51:           if possibleNewGroup.Count == 1 then
52:             allEdgesUsed = true
53:           else
54:             edges = possibleNewAdj
55:             currentGroup = possibleNewGroup
return groups

```

Implementação

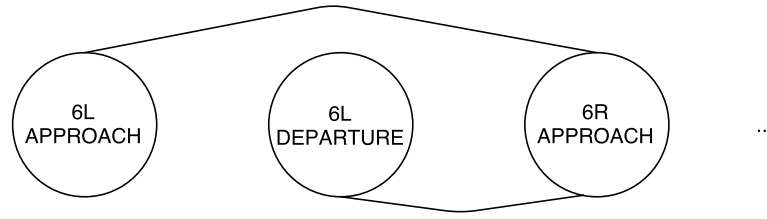


Figura 5.5: Exemplo de uma ligação no grafo mostrando que os as pistas 6L *Approach* e 6R *Approach* podem ser usadas em simultâneo.

o conjunto de pistas ativo, se necessário, de modo a permitir um maior fluxo de aeronaves. Sendo que para a determinação do melhor grupo é preciso ter em atenção vários atributos, recorreu-se ao *SAW (Simple Additive Weighting)* como algoritmo de decisão, onde a cada atributo é dado um peso e onde a soma de todos os pesos deve ser igual a 1. É possível ver na Equação 5.1, a forma como é calculado para cada grupo a sua eficiência caso seja escolhido, onde *ac*, *ar*, *dc* significam, respetivamente, número de pistas ativas, número de pistas a serem usadas para aterragem e número de pistas a serem usadas para partidas. Enquanto que *K_r* tem um peso constante ao longo das diferentes demandas, *K_{app}* e *K_{dep}* variam de acordo com as diferentes necessidades do aeroporto. Antes de serem normalizados para que a soma entre todos os pesos seja iguala uma unidade, *K_{app}* e *K_{dep}* correspondem aos coeficientes entre o número de voos para aterrar ou para partir e o número total de operações existentes no intervalo de tempo a considerar

$$P_{grouprunway} = K_r \frac{cur_{ac} - worst_{ac}}{best_{ac} - worst_{ac}} + K_{app} \frac{cur_{ar} - worst_{ar}}{best_{ar} - worst_{ar}} + K_{dep} \frac{cur_{dc} - worst_{dc}}{best_{dc} - worst_{dc}} \quad (5.1)$$

Por fim, ainda se implementou um dos fatores mais importantes para a escolha das pistas - a direção do vento. Após a realização de uma breve pesquisa, optou-se por utilizar a *API* fornecida pela plataforma *openWeatherMap* visto que o serviço permitia obter informações dada uma localização geográfica (latitude, longitude). De modo a conseguir obter informações atualizadas sobre as condições atmosféricas presentes nos aeroportos, foi criado um mecanismo baseado em *Threads* que de forma cíclica provocam a execução de uma função criada declarada no ficheiro *gmaps.html* retornando assim os dados necessários. Embora a plataforma já disponha das informações meteorológicas para os vários aeroportos, essa informação não está a ser usada para determinar o grupo de pistas ativas visto que os dados usados para os testes reportam-se ao passado onde é impossível aceder aos dados existentes na altura.

5.3.2 Gestão das aeronaves em espera

Sendo necessário organizar o espaço aéreo nas proximidades do aeroporto, era fundamental a construção de um módulo capaz de controlar as aeronaves que se encontrassem em fase final do seu voo. Embora uma forma simplificada de organização do espaço aéreo já existisse, a mesma não assegurava nem a segurança das aeronaves nem algumas das regras impostas pela *FAA*. O módulo desenvolvido no decorrer desta dissertação destaca-se do seu antecessor pelas mesmas razões que

Implementação

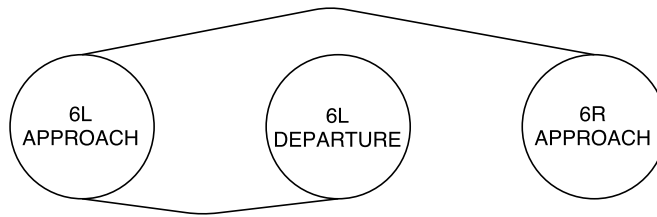


Figura 5.6: Exemplo de um caso possível que levaria à criação de dois grupos.

se aproxima da realidade, uma melhor gestão interna (existência de níveis assim como a descida entre os mesmos) e externa da fila (o balanceamento das várias filas), uso das várias entradas vistas na secção 2.3.1 e um cálculo mais realista do percurso das aeronaves enquanto executam a manobra de espera. À data do início da implementação da manobra de espera, a mesma não fazia respeitar as regras impostas pela entidade controladora do espaço aéreo norte-americano, tal como expostas na secção 2.3.2. Uma das primeiras tarefas a realizar foi necessariamente o forçar que estas regras eram cumpridas e como tal, foi necessário alterar a distância a percorrer pelas aeronaves em função da sua velocidade assim como alterar a velocidade dependendo da altitude onde a manobra se irá realizar. Com o intuito de aproximar a plataforma à realidade era necessário, como visto na Tabela 5.1, que a plataforma conseguisse executar ambos os padrões de manobras de espera - *standard* e *non standard*, sendo que para tal era necessário alterar a forma como os pontos eram calculados.

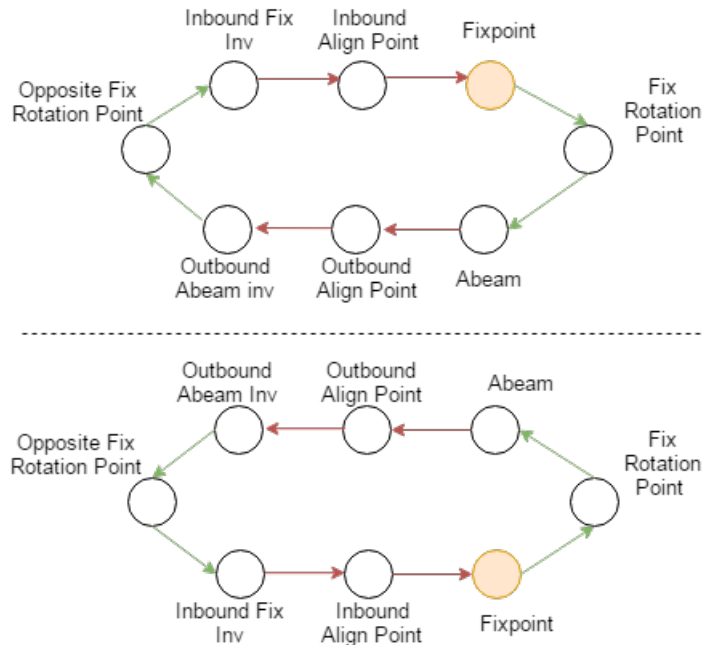


Figura 5.7: Exemplo da disposição dos pontos entre uma manobra de espera *standard* e *non standard*.

Sendo então uma necessidade permitir a execução de um manobra *non standard* foi necessário modificar o conteúdo da mensagem enviada pelo agente ATC ao agente veículo, como será

descrito mais à frente. Com as informações necessárias à execução da manobra, era necessário modificar a distância que a aeronave iria percorrer entre *legs* - *Inbound* e *Outbound* assim como entre os semi-círculos visto que as mesmas eram sempre uma constante. É relevante informar que as várias partes do padrão, *Inbound* e *Outbound leg* e os semi-círculos, têm como principal unidade de medida o tempo, ou seja, diferentes aeronaves com diferentes altitudes e velocidades, devem percorrer a mesma parcela na mesma duração. Com isto, e tendo em conta as regras referenciadas na secção 2.3.2, a aeronave com base na altitude na qual irá realizar a manobra e com base na sua velocidade calcula a distância a percorrer entre as várias partes constituintes da manobra. É possível visualizar através da Figura 5.7 a quantidade de pontos gerados após o agente aeronave ter recebido o *fixPoint*. Embora a arquitetura da manobra já estivesse calculada era ainda necessário que a aeronave seguisse as recomendações da *FAA* para a entrada na mesma. Foi então necessário determinar, com base na posição da aeronave e direção da manobra, a melhor entrada segundo a entidade reguladora, sendo que as possibilidades podem ser revistas na secção 2.3.1. A possibilidade de realizar esta tipo de manobras mostra-se incompleta quando comparada com o que existe na realidade, visto que não assegura uma gestão segura e organizada das aeronaves no espaço aéreo. Como forma de resolver a inexistência de uma organização racional das aeronaves num local próprio para a realização de manobras de espera assim como da conectividade entre os vários locais existentes no espaço aéreo foi necessário o desenvolvimento de uma lógica capaz de albergar toda essa responsabilidade. Esta lógica tem início aquando o carregamento do aeroporto e das respetivas rotas de aproximação para cada pista, visto que é neste ultimo passo que são lidos os pontos onde se encontrarão definidos os pontos da manobra. De forma a organizar as várias aeronaves no espaço aéreo são criados vários "níveis" intervalados entre si de 1000ft de altitude, sendo que o número de níveis é pré-determinado na altura da criação dado o limite superior do espaço aéreo do aeroporto.

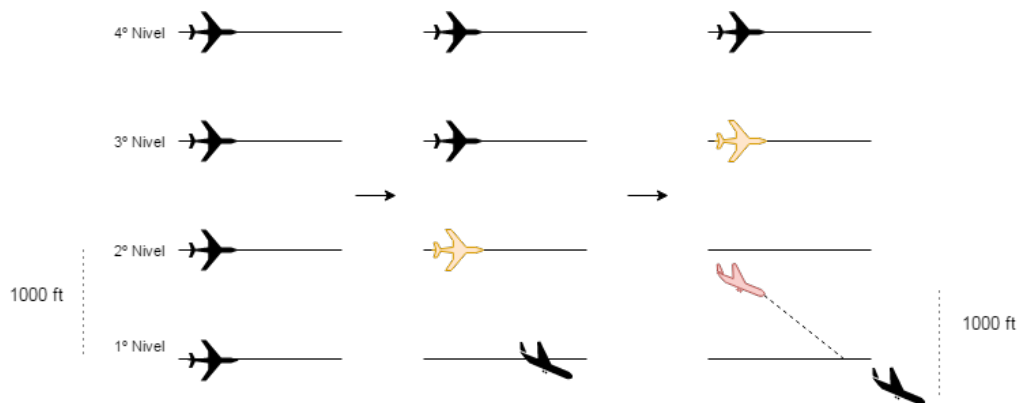


Figura 5.8: Exemplo da lógica após saída de uma aeronave do local de espera.

Assemelhando-se à realidade, uma aeronave após ver recusada a permissão para aterrar é re-direcionada para o nível mais inferior que se encontre livre e vai, com as saídas das aeronaves abaixo de si, descendo de nível até que é a sua vez de iniciar a aproximação final. Através da figura 5.8 é demonstrado a forma como a gestão interna deste módulo funciona, ou seja, o percurso das

aeronaves é sempre descendente e nunca ascendente e só é dada permissão para aterrar às aeronaves que se encontrem no primeiro nível. De forma a assegurar sempre a separação entre aeronaves foi necessário introduzir dois novos estados, *LAND_HOLD_PATTERN_AWAIT_DESCEND* e *LAND_HOLD_PATTERN_DESCEND*, aos que já existiam anteriormente (ver mais detalhadamente na secção 5.3.2).

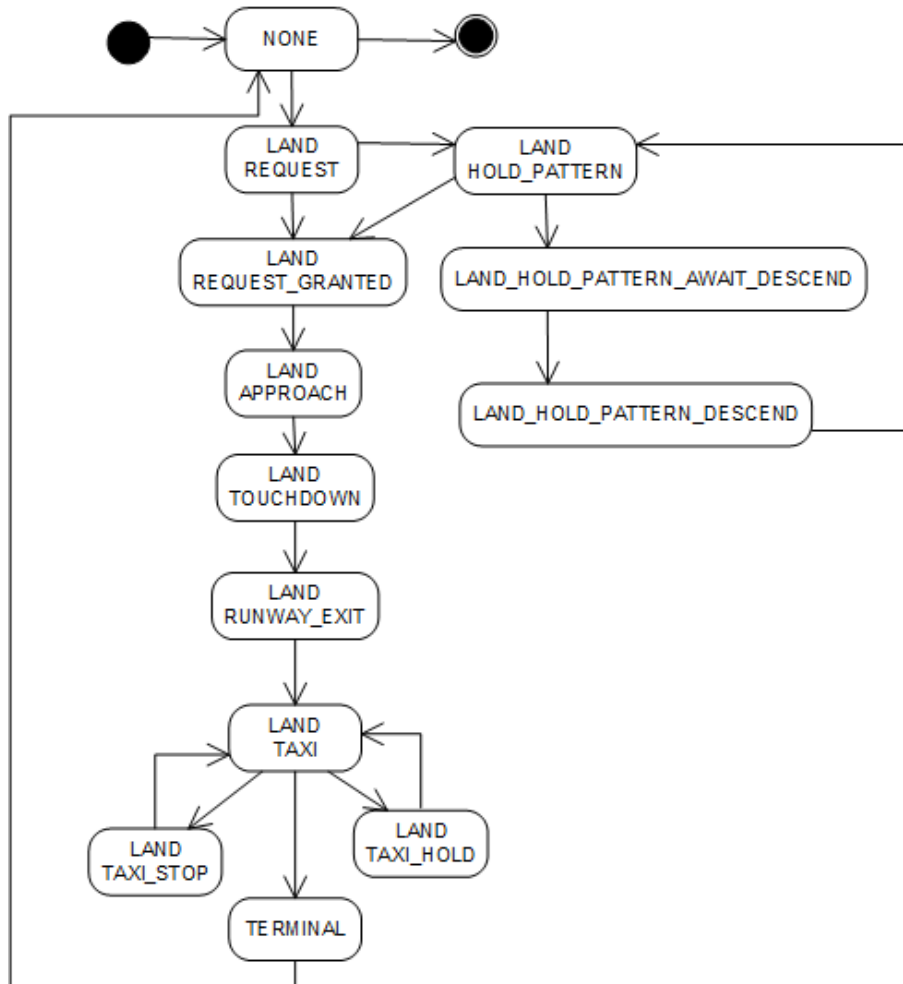


Figura 5.9: Máquina de estados de uma aeronave já com os novos estados adicionados.

Assim que o controlador de tráfego aéreo encarregue pela gestão da fila informa a aeronave presente no primeiro nível que pode dar início à aproximação à pista, o mesmo informa a aeronave imediatamente acima da primeira para aguardar por permissão para descer de nível, mudando o estado da aeronave de *LAND_HOLD_PATTERN* para *LAND_HOLD_PATTERN_AWAIT_DESCEND* (ver aeronave a cor de laranja no segundo esquema em 5.8). Com o libertar do nível por parte da aeronave a aterrar, o ATC informa o agente em espera que pode dar início à descida, mudando assim o seu estado para *LAND_HOLD_PATTERN_DESCEND* que será mantido até a aeronave completar a mudança de nível. Esta mudança de estado irá, consequentemente, libertar novamente o nível onde se encontrava a segunda aeronave. De forma a ocupar o mesmo, o controlador informa, caso exista, a aeronave no nível acima, acabando por alterar o estado da aeronave para

LAND_HOLD_PATTERN_AWAIT_DESCEND. Este ciclo repete-se até que todas as aeronaves terminem a descida de pelo menos um nível da fila. Com a implementação dos novos estados e com a consequente necessidade de sincronizar os vários agentes envolvidos foi necessária a criação de novas mensagens fazendo assim alterar o fluxo de mensagens para aeronaves em *hold*. Na figura 5.10 está descrito o fluxo das mensagens trocadas entre o controlador ATC e os vários agentes veículo. Às mensagens já existentes, *holdFix* e *clearToLandRunway*, ambas alteradas no decorrer desta dissertação, foram adicionadas as mensagens *changeHoldPatternAltitude*, *startDescendOnHold* e *finishDescend*.

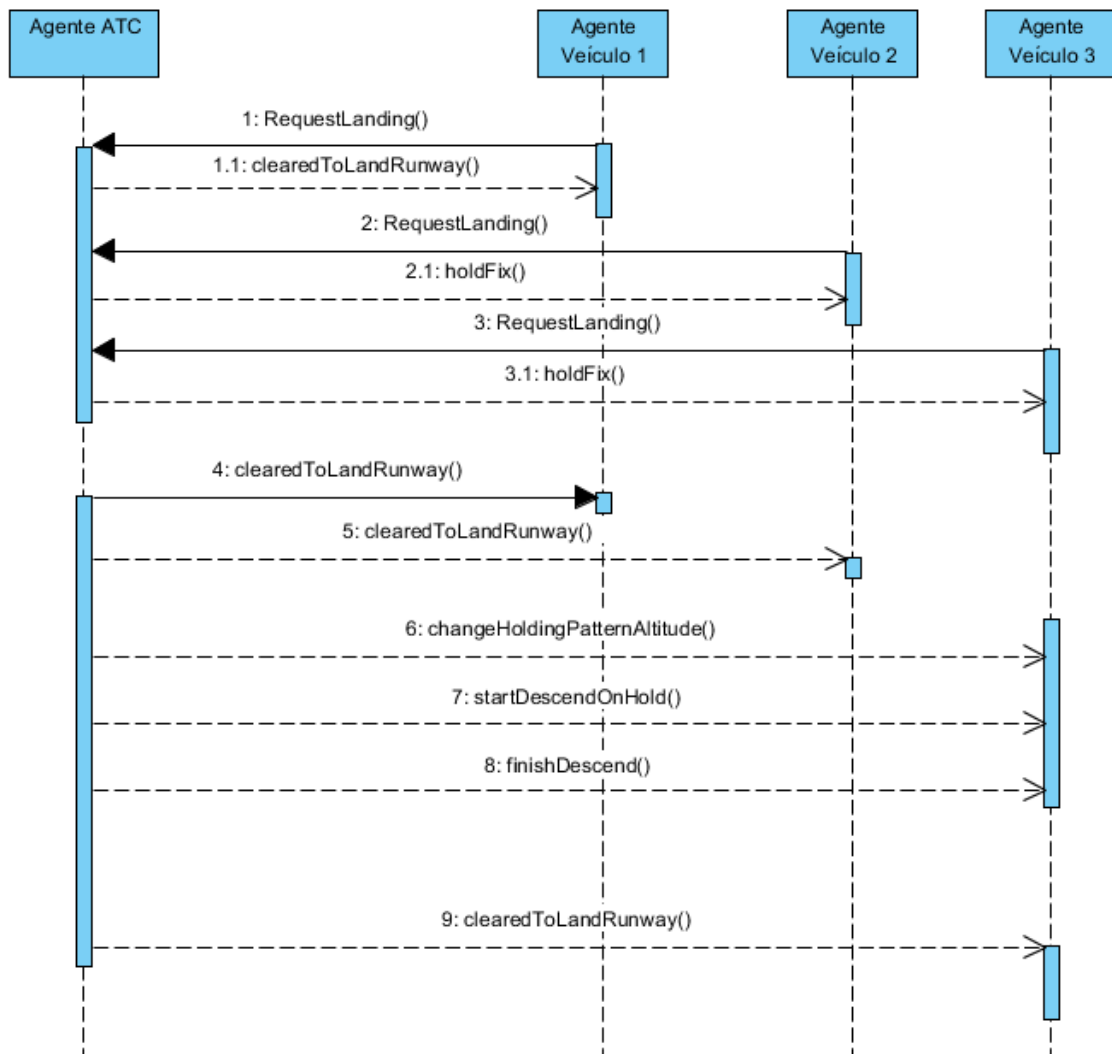


Figura 5.10: Exemplo de uma possível troca de mensagens entre agentes veículos e um agente ATC.

Ao ver o pedido para aterragem negado, a aeronave é redirecionada para um ponto de espera através da mensagem *holdFix* cujo conteúdo foi alterado de forma a transmitir, para além das coordenadas com altitude e a direção sobre a qual a manobra deve ser feita, se a manobra é *standard* ou

non standard. Para além da primeira alteração, modificou-se ainda a mensagem que dá permissão à aeronave para aterrar de forma a que a aeronave receba o conjunto de pontos referentes à aproximação à pista. Com o objetivo de iniciar a descida de nível na fila de *holding*, o *ATC* envia uma mensagem *changeHoldPatternAltitude* onde é transmitido a nova altitude para a qual a aeronave se deve deslocar, mudando assim o estado da aeronave quer no *ATC* quer no próprio agente do veículo para *LAND_HOLD_PATTERN_AWAIT_DESCEND*. Ao ser possível a descida à altitude desejada por parte da aeronave, o *ATC* informa a mesma enviando uma mensagem *startDescendOnHold*. O estado consequente da mensagem anterior, *LAND_HOLD_PATTERN_DESCEND*, apenas é alterado para *LAND_HOLD_PATTERN* aquando do envio de uma nova mensagem, *finishDescend*, por parte do *ATC*. De forma a conseguir gerir a carga das várias filas de espera no espaço aéreo assim como a diminuição do tempo que uma aeronave aguarda por permissão por aterrar foi criada uma fórmula matemática que se pode ver na Equação 5.2, responsável por atribuir um determinado ponto de espera a uma aeronave.

$$T_i = T_{travel}^i + T_{estimated_hold}^i + Penalty^i \quad (5.2)$$

Após a entrada no espaço aéreo e consequentemente o pedido de permissão para aterrar, o controlador faz uma breve análise às várias filas de espera ativas de forma a melhorar o fluxo de aeronaves. Como forma de avaliação, o controlador recorre ao uso de três variáveis sendo os mesmos, com exceção de *Penalty*, representativos do tempo que uma aeronave iria dedicar nos vários momentos da sua manobra de espera. O primeiro atributo a ser calculado, T_{travel} , representa a duração da viagem entre a localização atual da aeronave ao *holding fix* considerando que a velocidade da mesma seria mantida constante. Por outro lado, o segundo atributo, $T_{estimated_on_hold}$, baseia-se na quantidade de aeronaves existentes na fila assim como duração *holding fix - pista*, média do tempo esperado pela permissão para aterrar, como forma de conseguir estimar o tempo a ser usado pela aeronave enquanto a mesma espera por permissão para aterrar. Aos dois primeiros atributos adiciona-se uma penalização binária [zero ou *Infinito*] sendo que em caso da fila se encontrar em capacidade máxima o valor deste atributo adota o valor de nove mil. Esta capacidade máxima é calculada pelo controlador de tráfego aéreo no início da simulação tendo em conta as limitações do espaço aéreo pelo qual está responsável, $Altitude_{airport_airspace}$, assim como a altitude do primeiro nível da fila, $Altitude_{minimum_holding_altitude}$.

$$C_{maxima} = \frac{Altitude_{airport_airspace} - Altitude_{minimum_holding_altitude}}{1000} \quad (5.3)$$

Após o cálculo de T_i para as filas de espera ativas é atribuída a fila cujo tempo total esperado é menor.

5.4 Criação de um novo controlador

Em versões anteriores da plataforma, as simulações encontravam-se confinadas a um único aeroporto e ao respetivo espaço aéreo e, como tal, a existência de um único tipo de controlador

Implementação

satisfazia as necessidades existentes. Com o decorrer da presente dissertação, e sendo um dos objetivos a gestão do tráfego aéreo entre aeroportos, era impreterível a criação de um novo tipo de controlador. *Tracon*, o nome dado ao novo tipo de controladores, têm como principal função o acompanhamento das aeronaves desde o momento em que abandonam o espaço aéreo do aeroporto de origem até à sua entrada no espaço aéreo do aeroporto de destino.



Figura 5.11: Exemplo da aérea controlada pelo controlador do tipo *Tracon* mapeado.

Sendo o novo controlador responsável por uma vasta área, atualmente mapeada para representar a zona dos grandes lagos (ver Fig.5.11), que se sobrepõe às várias zonas controladas pelos aeroportos, foi necessário a implementação de um procedimento capaz de determinar que controlador tem como responsabilidade a gestão de uma aeronave. Tendo conhecimento de que as aeronaves tendem apenas a entrar no espaço aéreo de um aeroporto caso o mesmo seja o seu destino ou origem, a atribuição do controlador tem por base um sistema de prioridades aliado à posição da aeronave. Num caso base, a aeronave apenas se encontra dentro dos limites estabelecidos de um espaço aéreo, facilitando assim a escolha do controlador responsável pela sua gestão. Contudo, é possível ocorrer que uma aeronave encontra-se inserida em múltiplos espaços aéreos. Neste cenário, respeitando o facto que um controlador de aeroporto possui uma prioridade mais elevada em relação ao controlador *Tracon*, é atribuída a gestão da aeronave ao controlador que tiver uma prioridade mais elevada. Dado o objetivo que levou à implementação desta característica é natural que uma aeronave precise de transitar entre diferentes espaços aéreos geridos por diferentes controladores. Com o objetivo de replicar o que acontece na realidade foi criado um sistema de mensagens capazes de alertar os vários controladores da entrada e saída das aeronaves no espaço aéreo.

Na figura 5.12 está demonstrado o fluxo de mensagens no momento em que a mudança de controlador é necessária. Em constante monitorização do espaço aéreo, o agente veículo ao se aperceber que entrou num espaço aéreo cujo controlador possui maior prioridade que o atual, envia uma mensagem do tipo *InformExitAirspace* ao controlador responsável pelo espaço aéreo a

Implementação

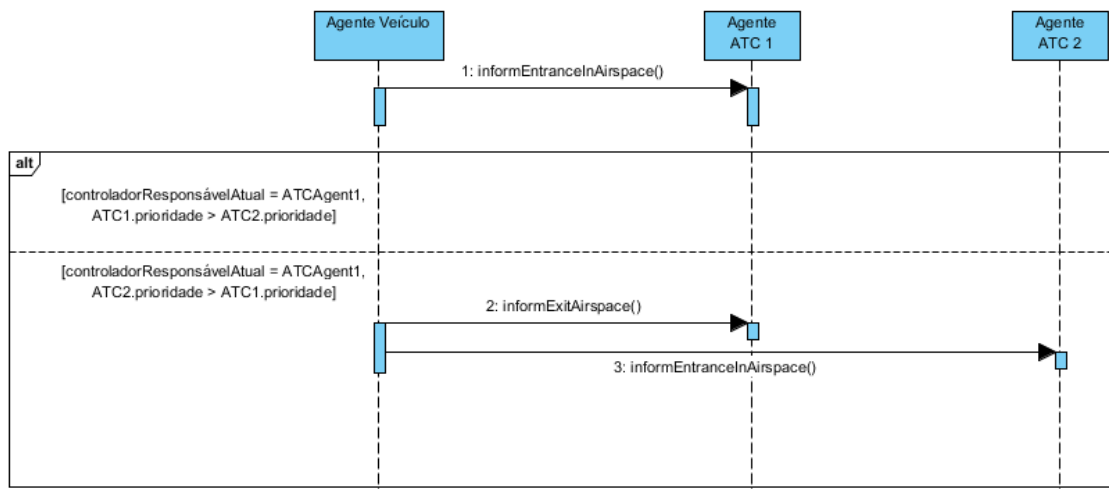


Figura 5.12: Exemplo de mensagens trocadas num cenário em que a aeronave transita do espaço aéreo do controlador *ATC 1* para o *ATC 2*

deixar. Após o envio desta mensagem, o veículo informa o seu novo controlador que se encontra sobre a sua gestão através do envio da mensagem *InformEntranceInAirspace*.

Sendo responsável pelo controlo das aeronaves entre aeroportos era necessário que o controlador do tipo *tracon*, à semelhança do seu homólogo responsável pela gestão dos aeroportos, tivesse conhecimento completo do seu espaço aéreo. De forma a obter este conhecimento omnisciente foi necessário a inclusão de informação *XML* que não se encontrava em uso embora já se encontrasse criada, visível na Listagem 5.3, contendo os vários nós do espaço aéreo (*waypoint*) assim como as vias de circulação entre os vários nós (*airway*). Aproveitando a estrutura da informação *XML*, o espaço aéreo foi implementado como uma grafo bidirecional onde os nós apresentam-se com dois atributos, nome e coordenadas, e as adjacências entre nós contêm apenas a origem e o destino daquela via deixando por utilizar o atributo *trafficedensity*, representativo da utilização, em percentagem, daquela via quando comparada com todas as outras adjacências com o mesmo nó de origem.

```

1  <airspace>
2    <waypoints>
3      <waypoint id="w1" name="KMSP">
4        <latitude>44.882</latitude>
5        <longitude>-93.2218</longitude>
6      </waypoint>
7      ...
8    </waypoints>
9    <airways>
10     <airway id="aw1-2">
11       <origin>1</origin>
12       <destination>2</destination>
13       <trafficedensity>14.02</trafficedensity>

```

Implementação

```

14         </airway>
15         ...
16         <airways>
17     </airspace>

```

Listagem 5.3: Exemplo da estruturação dos dados relativos ao espaço aéreo

Após o mapeamento do espaço aéreo das aeronaves e com a finalidade de conseguir impedir, caso necessário, aeronaves de congestionar as áreas geridas pelos aeroportos, foi feito um levantamento dos vários *STAR* de forma a se conseguir detetar os vários pontos onde a realização de manobras de espera é possível. Com isto e à semelhança das rotas de aproximação na secção 5.2, foram acedidos os documentos oficiais para os aeroportos mapeados na plataforma de forma a se preencher uma tabela para que posteriormente a informação fosse transcrita para formato *XML*.

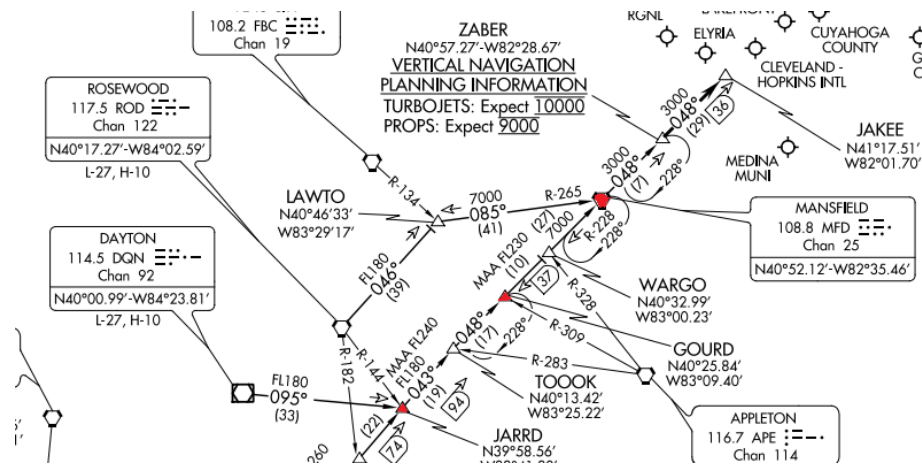


Figura 5.13: Rota de um dos *STARs* com manobras de espera conectadas entre si.

Através de cartas semelhantes à representada na Figura 5.13 foram extraídos atributos como a localização dos vários pontos, nome e informações relativas a filas de espera. Em relação às localização foram retirados três atributos (latitude, longitude e altitude) sendo os três igualmente importantes de modo a replicar a realidade na plataforma. De modo a que o controlador consiga sequenciar as aeronaves ao longo das várias filas de espera foi necessário a criação de um atributo "Ligação de". Por fim, foram recolhidas as informações relativas à direção e ângulo da manobra.

Tabela 5.3: Exemplo dos vários pontos perto do aeroporto de Cleveland, OHIO onde se é capaz de realizar manobras de espera.

STAR	Latitude	Longitude	Altitude	Ligação de	Sequência	Nome	Âng. da manobra	Direção
HIMEZ	41° 50' 58.91"N	82° 12' 04"W	7000				382	L
KEATN	40° 55' 53.03"N	81° 39' 20.12"W	3000				322	L
ZABER	41° 17' 30.83"N	82° 01' 42.00"W	3000	MANSFIELD			48	R
	40° 52' 7.2012"N	82° 35' 27.60"W	7000	GOURD	1	MANSFIELD	48	R
	40° 25' 50.33"N	83° 09' 24.22"W	7000		2	GOURD	48	R

Após a conversão da informação para formato *XML*, a mesma é introduzida no ficheiro de configuração de cenário com a estrutura representada em 5.4.

Implementação

```
1      <stars>
2          <star name="KEATN" holdingDegrees="322" holdingDir="L">
3              <coordinates>
4                  <latitude>40° 55' 53.03''N</latitude>
5                  <longitude>81° 39' 20.12''W</longitude>
6                  <altitude lengthUnit="Foot" measured="amsl">3000</altitude>
7              </coordinates>
8              <connections />
9          </star>
10         <star name="ZABER" holdingDegrees="48" holdingDir="R">
11             <coordinates>...</coordinates>
12             <connections>
13                 <courses>
14                     <course cid="0">
15                         <Fixpoint name="MANSFIELD" holdingDegrees="48" holdingDir="R">
16                             <coordinates>...</coordinates>
17                         </Fixpoint>
18                         <Fixpoint name="GOURD" holdingDegrees="48" holdingDir="R">...</
19                             Fixpoint>
20                     </course>
21                 </courses>
22             </connections>
23         </star>
24     </stars>
```

Listagem 5.4: Exemplo da estruturação dos dados relativos aos *stars* nas proximidades do aeroporto de Cleveland, Ohio

A inexistência de voos entre aeroportos aliada a ser apenas feita uma gestão das infraestruturas dos aeroportos assim como o seu espaço aéreo, justificava a falta de implementação de planos de voos. De forma a ser possível a implementação das rotas a serem usadas foi necessária o carregamento de informação que foi sendo estruturada em conjunto com os orientadores (ver código 5.5). Na sua base, uma rota é constituída por um conjunto de pontos (*route*) e um tipo de evento responsável por iniciar a viagem (*activationCondition*).

```
1      <strategy>
2          <strategyEntry>
3              <activationCondition>
4                  <trigger triggerType="Time">
5                      <phaseTimePoint countingFrom="absolute">
6                          <phaseDateTime>2015-11-04T04:06:00Z</phaseDateTime>
7                      </phaseTimePoint>
8                  </trigger>
9              </activationCondition>
10             <tactic>
11                 <action>
```

Implementação

```
12         <route>
13             <wpoint name="KMSP">
14                 <altitude measured="amsl" lengthUnit="Foot">841</altitude>
15             </wpoint>
16             <wpoint name="JEDET">
17                 <altitude measured="amsl" lengthUnit="Foot">12725.416666667</
18                     altitude>
19                 <after>PT00H07M26S</after>
20             </wpoint>
21             <wpoint name="ZMBRO">...</wpoint>
22             ...
23         </route>
24     </action>
25 </tactic>
26 </strategyEntry>
</strategy>
```

Listagem 5.5: Exemplo da estruturação de uma rota

Com a implementação das características mencionadas tornou-se possível a existência de rotas aéreas entre aeroportos seguindo de forma realística os planos de voos que existiam. No entanto, de modo a gestão das aeronaves que se encontram em voo fosse exequível, foi necessário aprimorar a comunicação entre os vários agentes envolvidos nesta tipo de operação. Com isto foi necessária a criação de um novo fluxo de informação através de mensagens no sistema. Sendo que o controlador *tracon* é o tipo de controlador que mais margem para controlar o espaço aéreo possui, foi criado um procedimento capaz de o informar o estado de congestionamento de um aeroporto. Em situações atípicas em que um aeroporto não consegue gerir mais aeronaves dado o congestionamento do seu espaço aéreo, o controlador responsável por essa gestão envia *informFullCapacity(aeroportoID, atraso)* onde o último parâmetro é calculado pela subtração do horário previsto de aterragem ao horário real de chegada. Ao receber esta informação, o *Tracon* impede que as aeronaves com destino a esse aeroporto transitem de espaço aéreo, ordenando que as mesmas façam manobras de espera nos pontos vistos na Tabela 5.3. Se ao fim do atraso informado pelo controlador do aeroporto, o controlador *tracon* não tiver recebido uma atualização do estado de congestionamento o mesmo envia uma nova mensagem, *requestClearanceAfterDelayTime*, a requerer nova informação da situação. Caso o gestor do espaço aéreo do aeroporto, após envio da mensagem a informar que se encontra em capacidade máxima, repare que a sua capacidade desceu para um determinado valor de pontos percentuais, o mesmo envia uma mensagem a informar o *tracon* que se encontra disponível para receber mais aeronaves. Em situações extremas o controlador *tracon* compromete-se a atrasar as várias aeronaves com destino a aeroportos em stress, sendo que para tal recorre a métodos como redirecionamento, alteração de velocidades e de horários de partida das várias aeronaves.

De forma a facilitar o controlo das várias aeronaves foi criado uma nova máquina de estados contendo cinco estados - *EN_ROUTE*, *ON_HOLD*, *ON_HOLD_PATTERN_AWAIT_DESCEND*, *ON_HOLD_PATTERN_DESCEND* e *DELIVERING*.

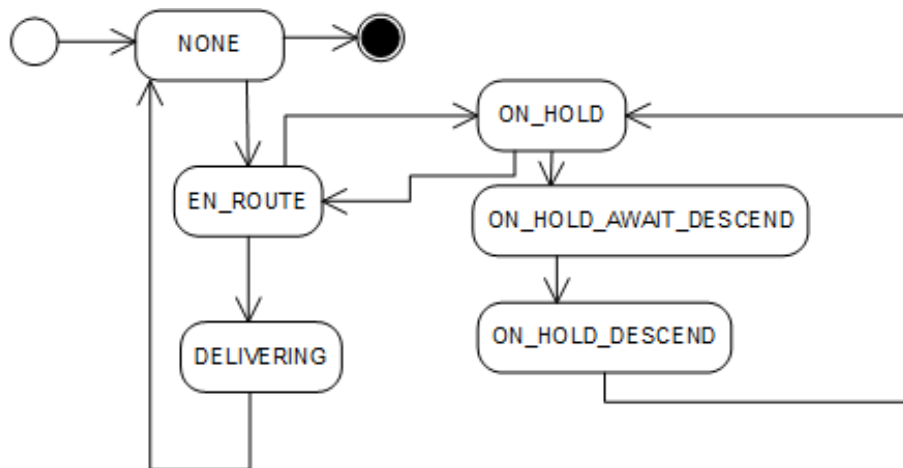


Figura 5.14: Máquina de estados para as aeronaves geridas pelo novo controlador.

Tal como é observado na Fig.5.9, ao reconhecer que uma aeronave se encontra dentro do seu espaço aéreo, o controlador de tráfego aéreo atribui-lhe o estado *NONE*. Este estado é apenas alterado para *EN_ROUTE* aquando da receção de uma mensagem por parte da aeronave a informar que o controlador é o responsável pela sua gestão. O estado *EN_ROUTE* representa que a aeronave se encontra a caminho do seu destino até que quando se encontrar a três quilómetros do último ponto do plano de voo, o controlador decide se a aeronave pode, de facto, transitar para o espaço aéreo do destino (*Delivering*) ou se é necessário redirecionar a aeronave para um fila de espera. À semelhança dos estados *LAND_HOLD_PATTERN_AWAIT_DESCEND* e *LAND_HOLD_PATTERN_DESCEND* vistos na secção 5.3, também a criação e uso dos estados *ON_HOLD_AWAIT_DESCEND* e *ON_HOLD_DESCEND* têm como objetivo assegurar a distância mínima obrigatória entre aeronaves.

À semelhança do controlador aéreo responsável pela gestão de zonas aeroportuárias, foi necessário a introdução de uma versão sequencial para a saída das aeronaves da fila de espera. Dado que, com a introdução desta versão, múltiplas aeronaves poderiam encontrar-se em direção aos aeroportos, foi impreterível a criação de um algoritmo (ver Algoritmo 2) capaz de prever as localizações das aeronaves em vários instantes t de forma a assegurar que a distância de segurança entre as aeronaves era sempre cumprida. Sendo esta distância de segurança uma variável que depende principalmente da ordem dos tipos de aeronave foi obrigatório a categorização das aeronaves em três tipos (leve, media e pesada) e a obtenção das distâncias mínimas a existir para os vários tipos de sequência.

5.5 Alteração das Linguagens Utilizadas

Sendo necessário um método capaz de definir com rigor as várias configurações usadas pela plataforma foram criados, em [Silva, 2011], quatro dialetos diferentes: *Scenario*, *Team*, *Missions* e *Disturbances*. Nesta secção são descritas as várias alterações realizadas nos dialetos provocadas pelas implementações referidas no presente capítulo.

Algorithm 2 Algoritmo responsável por permitir aeronaves saírem da fila de espera

```

1: procedure CANLEAVEHOLDING(AIRCRAFTDATA AIRC)
2:   Holding_fix_Point fp = getHolding(airc)
3:
4:   levelOfAircraft = fp.getCurrentlevelOfAircraft(airc) == 0
5:   isAirportAvailable = airportIsAvailable(airc.Destination)
6:   distanceIsBeingHeld = false
7:
8:   airData = null
9:   for all (string, AircraftData) lastAircleaving in leavingAircraftFromHolding(airc) do
10:    if lastAircleaving.item1 == airc.Destination then
11:      airData = lastAircleaving.aircraft
12:
13:   if airData != null then
14:
15:     airData.estimatedLandingTime = estimateTimeToLand(airData)
16:     currEstimatedLandingTime = estimateTimeToLand(airc)
17:
18:     currentdistNeededBetweenAircrafts = getDistance(airData.type, airc.type)
19:     distNeededBetweenAircrafts = distanceOfSeparation(airData.type, airc.type)
20:     timeInMinutes = distNeededBetweenAircrafts / airc.cruiseSpeed
21:
22:     if currentdistNeededBetweenAircrafts >= distNeededBetweenAircrafts &&
currEstimatedLandingTime > airData.estimatedLandingTime + timeInMinutes then
23:       distanceIsBeingHeld = true
24:     else
25:       distanceIsBeingHeld = true
26:
27:   return levelOfAircraft && isAirportAvailable && distanceIsBeingHeld

```

5.5.1 Scenario

Responsável pela configuração e definição do cenário a ser usado pela plataforma, o dialeto *SDL* (*Scenario Description Language*), abordado em [Silva et al., 2017], foi sofrendo alterações que se foram mostrando necessárias com o objetivo de acomodar a adição das implementações referenciadas no presente capítulo.

$$\begin{aligned}
 \text{Scenario} &= \langle B, C, T, N, A \rangle \\
 B &= \text{Base} \\
 C &= \text{Controller} \\
 T &= \text{AgentType} \\
 N &= \text{Area} \\
 A &= \text{Airspace}
 \end{aligned}
 \tag{5.4}$$

Uma das funcionalidades que obrigou à alteração do dialeto foi a introdução de um novo controlador aéreo. Dada a adição deste e tendo em consideração as suas respetivas funções mostrou-se necessário a introdução de um novo conjunto, *A* na Equação 5.4, capaz de definir um determinado espaço aéreo. Tal como é possível notar através da Figura 5.15, a secção do dialeto *Airspace* é constituído por dois conjuntos: rotas (*airways* na nomenclatura inglesa) e por pontos do espaço aéreo (*waypoints* na nomenclatura inglesa).

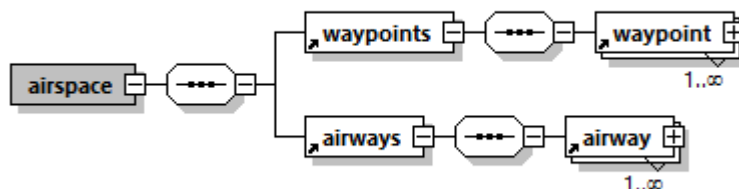


Figura 5.15: Representação gráfica dos atributos que constituem a secção *Airspace* na *SDL*.

Sendo representativo dos vários locais existentes no espaço aéreo, *waypoints* representa o elemento pai de um conjunto ilimitado de *waypoint*. De forma a representar os mesmos locais, como é possível ver na Fig. 5.16, um *waypoint* é definido por um nome oficial, *id*, e pelas suas coordenadas.

Por outro lado, *airways* tenta representar as várias ligações existentes entre os pontos definidos em *waypoints*. Assim, para cada ligação ou aresta, existe um ponto de origem (*origin*), um ponto de destino (*destination*) e a quantidade de tráfego, em percentagem, que segue esta rota aquando da sua chegada à origem (ver Fig. 5.17). De forma a não ser criada ambiguidade nos dados, tanto os campos *origin* e *destination* tomam os valores que foram definidos em *waypoints*. De forma semelhante, a aresta toma como valor de *ID* *aw*{ID do ponto de origem}-{ID do ponto de destino}.

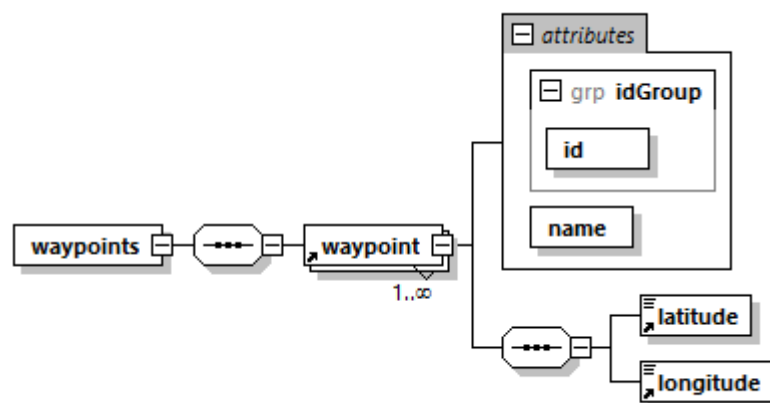


Figura 5.16: Representação gráfica do atributo *waypoints* na secção *Airspace*.

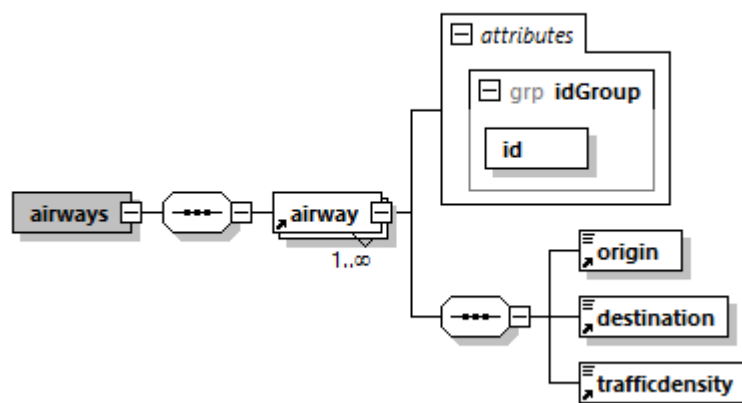


Figura 5.17: Representação gráfica do atributo *airway* na secção *Airspace*.

A introdução das várias rotas de aproximação, de partida e a criação dos *STARs* levou também à modificação da linguagem descritiva do cenário. Como forma de manter a informação organizada foi necessário a introdução de três novas componentes à linguagem.

- **Rotas de Aproximação**

Com o objetivo de mapear os vários pontos de aproximação existentes às pistas dos aeroportos foi necessário o desenvolvimento de um novo atributo *approachRoutes*. Este novo atributo mostra-se como uma sequência de *approachToRunway* em que cada elemento *approachToRunway* detém uma sequência de pontos *approachWaypoints* e *missedApproach* que representam a rota de aproximação e afastamento à pista, caso se falhe a aproximação à pista, respetivamente (ver Figura. 5.18).

Embora com nomes diferentes, ambas sequências são constituídas por zero ou mais *approachWaypoints*. Cada ponto de aproximação é constituído por quatro atributos sendo que ape-

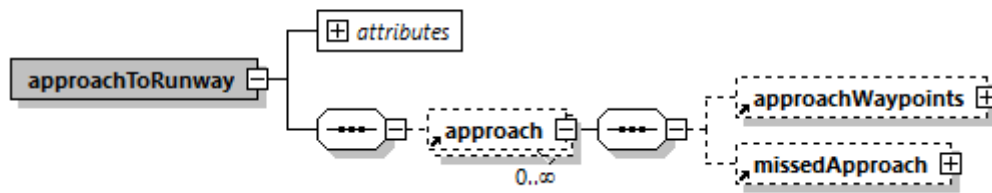


Figura 5.18: Descrição visual dos elementos existentes em *approachRoutes*.

nas dois podem tomar valores nulos (*holdingDegrees* e *holdingDir* no caso de *hasHolding* tomar o valor de *False*) e um elemento *coordinates*. Os atributos existentes encontram-se representados na Figura 5.19. O primeiro atributo define o nome do local de aproximação enquanto os restantes caracterizam a manobra de espera a ser realizada naquele ponto.

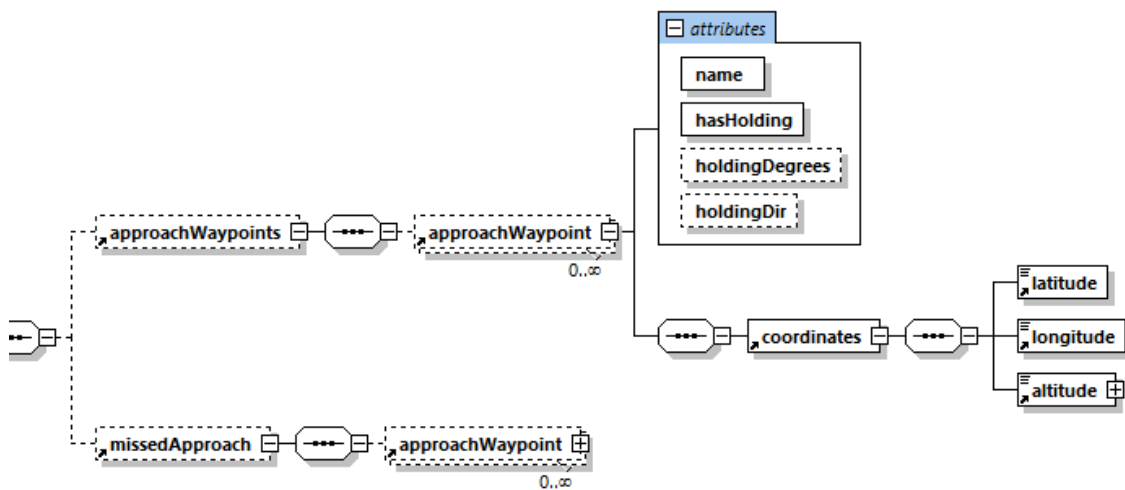


Figura 5.19: Representação gráfica dos atributos existentes em *approachWaypoint*.

• Rotas de Partida

Para a representação das várias rotas de partida existentes num aeroporto foram criadas componentes *departure* que se encontram adicionadas a componente-mãe *departures*. Cada *departure* contém informação relativa a uma rota de partida, sendo que a mesma pode ser partilhada entre várias pistas.

O elemento *fromRunway*, presente na Figura 5.20, é responsável por armazenar as pistas para a qual a presente rota diz respeito. Já *departureSequence* é um conjunto de *sequence-Alternative* que é constituído por um conjunto de *Fixpoint* (definidos por um atributo *name* e por um *coordinates*) ou por uma *altitude* (definidos apenas por uma *string* representativa da altura).

Implementação

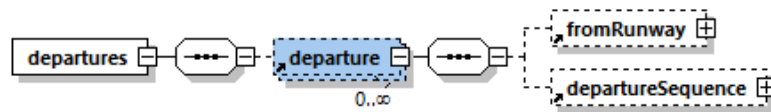


Figura 5.20: Representação da componente de *departures*.

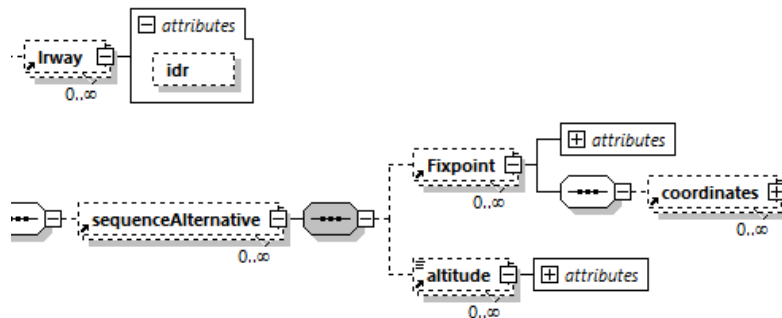


Figura 5.21: Representação das várias possibilidades de uma *sequenceAlternative*.

- **STARs**

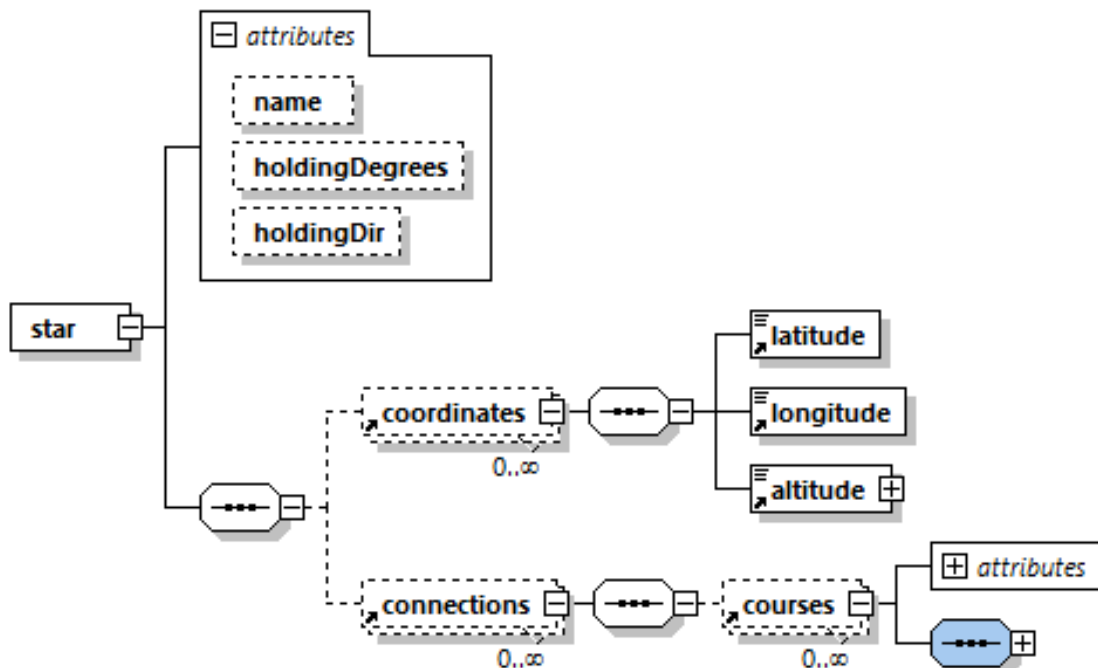


Figura 5.22: Representação da componente *star*.

De forma a representar os vários *STAR* foi necessário a criação de uma componente *stars*.

Cada elemento *star* possui um conjunto de atributos que definem a manobra de espera a ser realizada naquele local.

Sendo um *STAR* não só um local para a realização de manobras de espera, mas um lugar de ligação entre espaços aéreos foi necessário a inclusão de dois elementos, *connections* e *coordinates*, sendo este último responsável apenas pelo encapsulamento das coordenadas do ponto (ver Figura 5.22).

Representando as ligações existentes entre os *star* e os aeroportos, *connections* é uma sequência de *courses*. O elemento *courses* representa a ligação a um aeroporto sendo que o mesmo está definido através de um atributo. Dada a possibilidade de existirem múltiplas ligações ao mesmo aeroporto, foi criado um elemento, *course*, capaz de definir rotas de aproximação a um determinado aeroporto. Como é observável através da Figura 5.23, um elemento *course* é constituído por um atributo *cid* (*course ID*) e por um conjunto de *Fixpoint* que contém apenas um atributo *name* e um elemento *coordinates*.

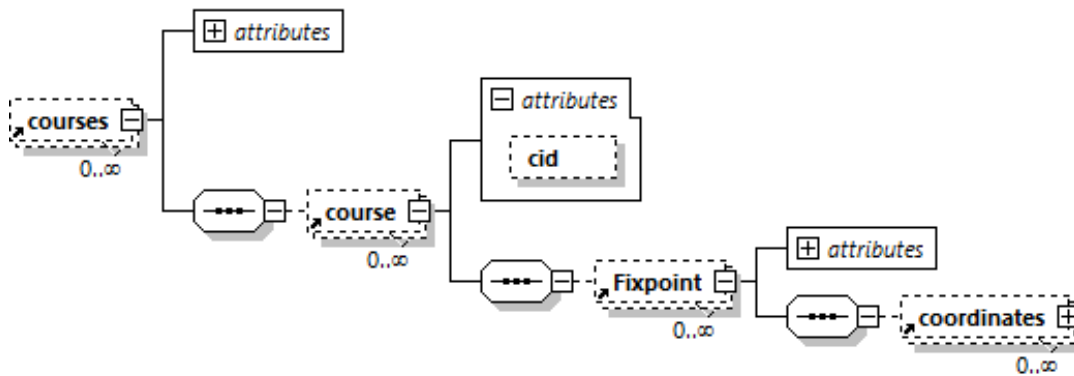


Figura 5.23: Representação da componente *courses*.

5.5.2 Mission

Com a introdução do novo controlador seguiu-se a inclusão das várias rotas que voos reais realizaram. De forma a introduzir essa informação para futuros testes, foi necessário a alteração do dialeto das missões *MDL* (*Mission Description Language*) abordado em [Silva et al., 2014].

De forma a tornar possível a inclusão das rotas foi então adicionado o elemento *route* como sendo um conjunto de *wpoint*. Cada *wpoint* é definido por um atributo *name* e um dois elementos *altitude* e *after*. Enquanto que o elemento *altitude* representa a altitude na qual a aeronave chegou ao local com nome *name*, o elemento *after* representa o tempo que a aeronave demorou desde o aeroporto de origem até ao ponto atual.

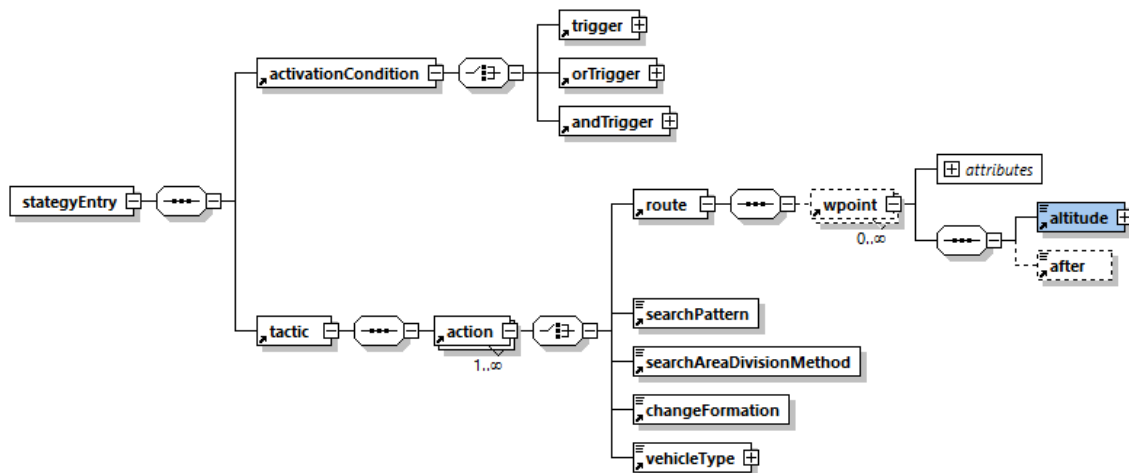


Figura 5.24: Representação da componente *courses*.

5.6 Alterações visuais nos vários componentes

Atualmente um conjunto de componentes da arquitetura da plataforma permitem a que o utilizador consiga alterar algumas variáveis de simulação. Contrariamente a este tipo de componentes, o controlador *ATC* para além dos requisitos necessários para comunicação entre os agentes da plataforma apenas permitia o carregamento de informação estática e a amostragem de variáveis que obtia da simulação. Com o objetivo de melhorar a experiência do utilizador, *UX*, e de permitir um melhor controlo do ambiente de simulação foram realizadas algumas mudanças visuais na componente do *ATC*.

De forma a facilitar ao utilizador possuir um maior controlo para o que o mesmo possa atuar mais rapidamente, achou-se necessário que a apresentação dos *logs* e das mensagens trocadas entre agentes deveria ser alterada. Comparativamente à versão anterior, a componente *ATC* para além de apresentar a informação num só separador, alterou o tamanho das janelas de forma a que se facilitasse a leitura de mensagens de tamanho elevado impedindo que as mesmas sofressem várias quebras de linhas.

Após a criação e distinção dos vários tipos de controladores (ver na secção 5.4) reparou-se que a janela onde era mostrado o espaço aéreo mostrava-se bastante pequena face ao tamanho da área controlada pelo novo tipo de *ATC*. Assim, manifestou-se interessante de aumentar de tamanho o mapa como forma a conseguir-se acompanhar um maior número de aeronaves sem que para isso fosse necessário recorrer à diminuição de zoom. Após a criação de um separador dedicado apenas à amostragem do espaço aéreo, e de forma a facilitar a obtenção de conhecimento do espaço aéreo assim como um melhor acompanhamento das rotas aéreas, foram implementadas opções capazes de adicionar informação no mapa. No caso do novo controlador ser do tipo *tracon* são agora mostrados os vários *waypoints* do espaço aéreo, como é visível na Fig. 5.25.

No caso do controlador *ATC* ser responsável pela gestão aérea nos aeroportos, ao invés de destacar os pontos pertencentes às rotas aéreas, é dada a possibilidade do mesmo ser capaz de,

Implementação

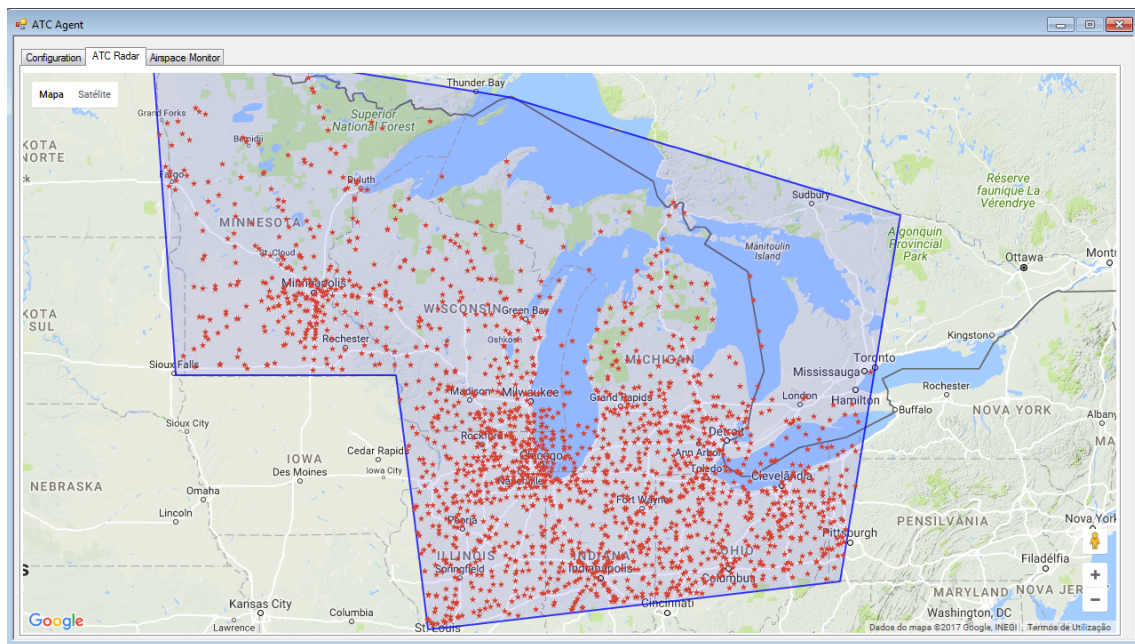


Figura 5.25: Representação do espaço aéreo para um controlador do tipo *Tracon*.

caso prefira, visualizar as várias rotas de aproximação assim como os vários pontos de espera presentes no seu espaço aéreo. Ainda para o mesmo tipo de controlador, e embora não seja usado nos casos de testes, foi dada a oportunidade ao utilizador de interagir com a simulação em tempo real. Como está exposto na Fig. 5.26, o utilizador pode decidir alterar o estado (Aberta/Fechada) de uma pista do aeroporto.

Com o intuito de melhorar a monitorização quer das aeronaves que ainda se encontram sob controlo dos controladores aéreos, foi adicionado à já existente tabela com múltiplas informações das aeronaves, uma tabela de aeronaves que já não se encontram no espaço aéreo. Esta tabela é atualizada sempre que o controlador receber uma mensagem por parte dos agentes *Vehicle* a informar que deixaram o seu espaço aéreo. Esta tabela é composta pela data de entrada no espaço aéreo, aeroporto de origem, data de saída do espaço aéreo e aeroporto de destino. (ver figura 5.26)

Ao longo da implementação de novas funcionalidades e dos respetivos testes intermédios era necessário o redirecionamento manual de várias aeronaves que se encontravam no espaço aéreo. Mostrando-se um processo que para além de moroso e custoso para a máquina onde eram lançados os agentes Veículo, foi fundamental a criação de um sistema no agente *Control Panel* capaz de comandar múltiplas aeronaves. Tal como demonstrado na figura 5.27 é possível ao utilizador fazer uma escolha detalhada das aeronaves que pretende controlar assim como escolher entre dois tipos de manobras diferentes.

De forma a que o *Control Panel* conseguisse comandar os vários agentes *Vehicle* foi necessária a criação de três mensagens diferentes, *controlPanel_asks_for_land*, *go_to_point_replace* e *go_to_point_add*. As duas últimas mensagens são responsáveis pela substituição/adição de pontos, respetivamente, à rota de uma aeronave. Ao contrário destas mensagens em que a opção do utilizador apenas leva ao envio de uma mensagem por aeronave presente no grupo a controlar,

Implementação

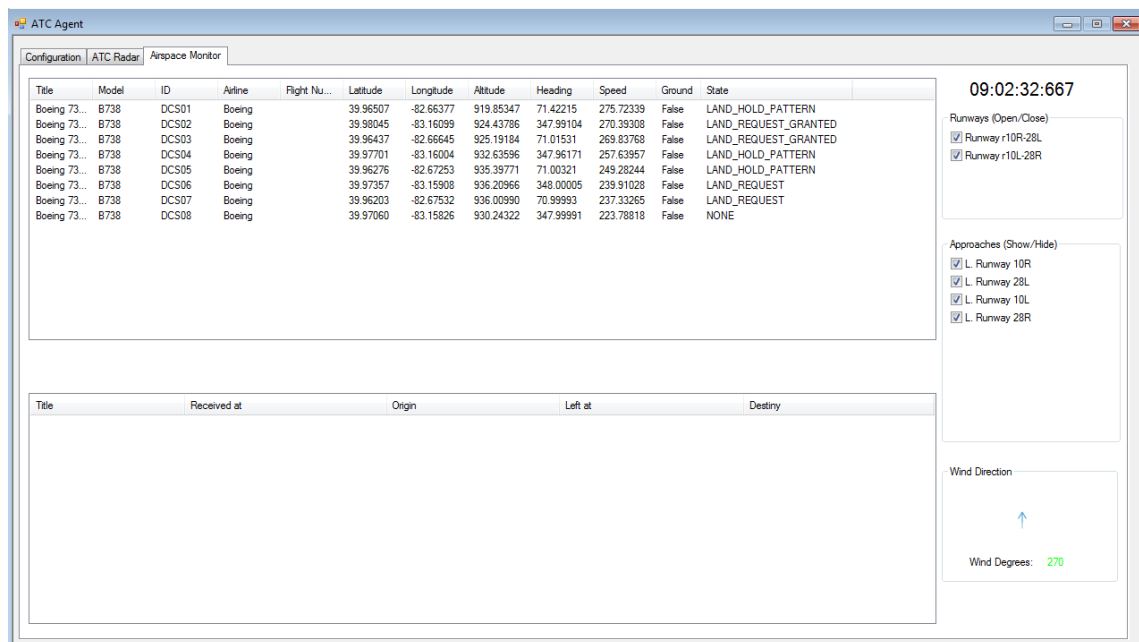


Figura 5.26: Exemplo de informações que podem estar representadas nas várias tabelas existentes.

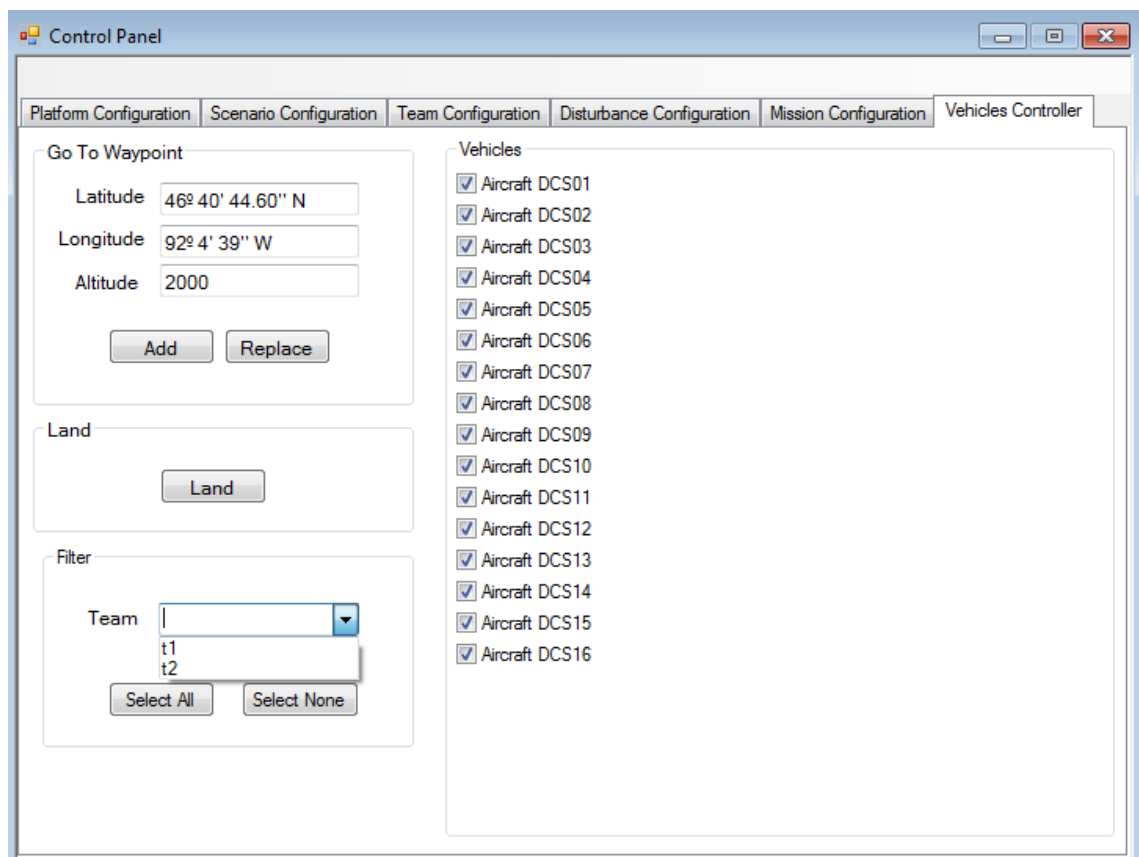


Figura 5.27: Novo separador contendo a funcionalidade de controlo de múltiplas aeronaves de forma simultânea.

Implementação

controlPanel_asks_for_land requer que o próprio veículo envie uma mensagem ao agente *ATC* para dar início ao procedimento de aterragem.

De forma a diminuir a necessidade de recorrer ao simulador de jogo para alterar variáveis da simulação foi ainda implementada a funcionalidade de poder alterar a taxa e a hora da simulação. Estas funcionalidades permitem uma maior facilidade no que diz respeito à realização dos testes.

Capítulo 6

Resultados

Serve o presente capítulo para a descrição e discussão dos vários testes realizados e os seus respetivos resultados.

6.1 Construção de Grupos de Pista

Na presente secção são detalhados os resultados obtidos para os vários casos aquando da criação de grupos de pistas.

6.1.1 Aeroportos Utilizados

Com o objetivo de testar o algoritmo de criação de pistas foi necessária a escolha de vários aeroportos de modo a que as suas infraestruturas servissem de base. Visto que o algoritmo utilizado faz uso das várias rotas de aproximação e de partida das pistas dos aeroportos, o conjunto de aeroportos passíveis de testes era constituído por apenas quatro aeroportos. No entanto, e apenas interessando a configuração das pistas, optou-se pela utilização dos aeroportos cuja configuração não possuisse semelhanças com as dos restantes aeroportos.

Para um primeiro cenário achou-se interessante escolher um aeroporto com a possibilidade de possuir pistas que pudessem ser usadas de forma paralela. Como tal optou-se pelo segundo aeroporto com maior número de passageiros no estado de Ohio, *KCMH (John Glenn Columbus International Airport)*. Tal como se pode observar na Fig. 6.1, o aeroporto possui duas pistas adjacentes sendo que a distância entre as mesmas é de aproximadamente 900 metros

Após a primeira escolha ter incidido numa configuração paralela de pistas, optou-se pela escolha de uma configuração diferente de pistas. Para tal recorreu-se ao aeroporto de *KCAK (Akron-Canton Regional Airport)*. O aeroporto usado num segundo cenário possui então, à semelhança do primeiro aeroporto, duas pistas no entanto as mesmas são pistas cruzadas. Na Fig. 6.2 é possível a visualização da disposição das pistas do aeroporto que serve como base para o segundo cenário.

Resultados

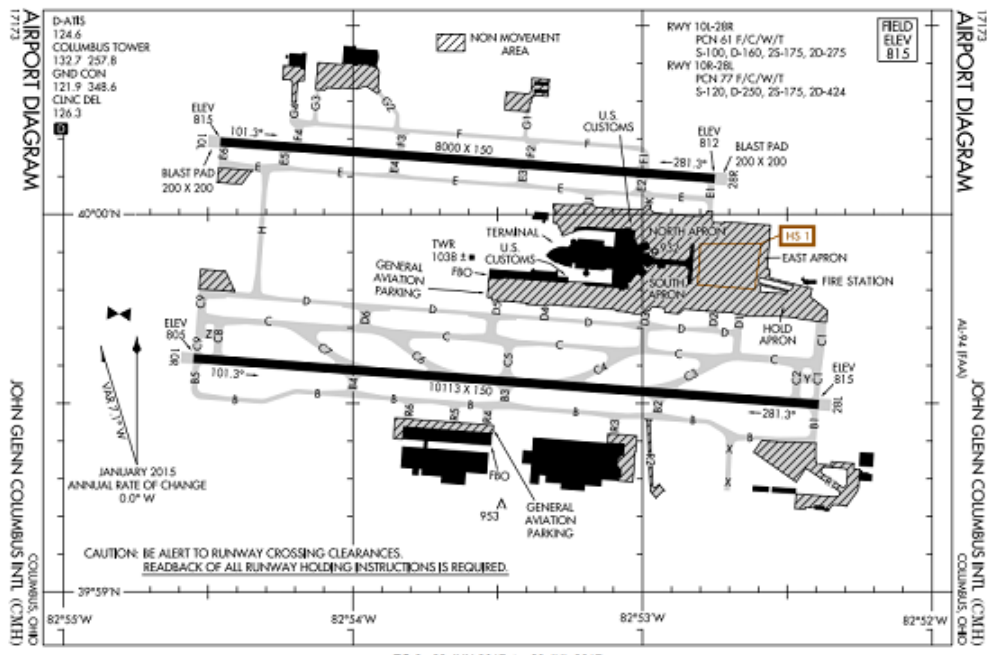


Figura 6.1: Configuração do aeroporto KCMH.

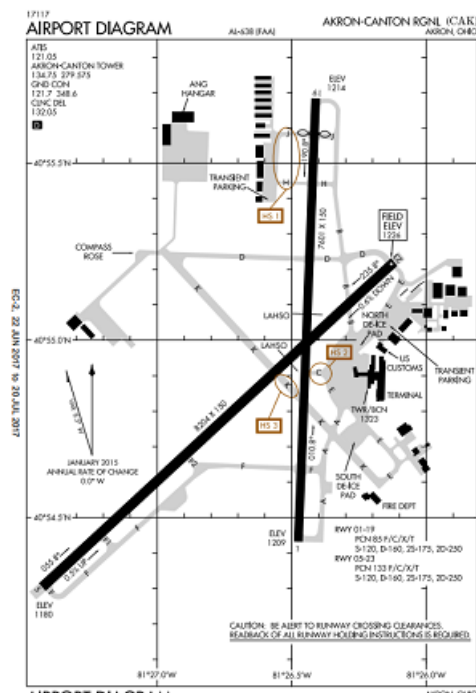


Figura 6.2: Configuração do aeroporto KCAK.

Para um terceiro cenário foi escolhido um aeroporto que possui mais que duas pistas. Dado que ambos os aeroportos escolhidos para os dois primeiros cenários encontravam-se no estado de Ohio, optou-se pela utilização do aeroporto KCLE (Cleveland-Hopkins International Airport) que se apresenta com um total de três pistas em que nenhuma é cruzada (ver Figura 6.3)

Resultados

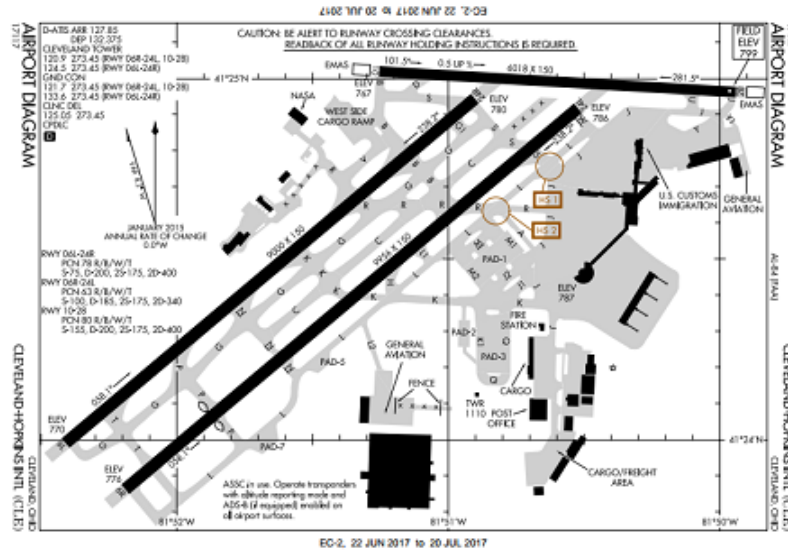


Figura 6.3: Configuração do aeroporto *KCLE*.

6.1.2 Resultados

Tendo como objetivo a avaliação do algoritmo de geração de grupos foram contabilizados quatro atributos, número de grupos criados, número de grupos só com pistas para chegadas ou partidas e, finalmente, grupos que contêm ambas. Na Figura 6.1 são demonstrados os resultados obtidos para os vários cenários, *KCMH*, *KCAK* e *KCLE* respectivamente.

Tabela 6.1: Resultados obtidos durante a criação de grupos

Cenário	Grupos Criados	Grupos Apr.	Grupos Dep.	Grupos Mis.
#1	12	4	4	4
#2	8	4	4	0
#3	12	6	6	0

Num cenário em que se pode utilizar pistas paralelas, como no caso do cenário 1, e após eliminação de grupos repetidos obteve-se um total de quatro grupos criados para cada atributo testado. Estes grupos resumem as várias direções pela qual as aeronaves podem executar as manobras pretendidas, isto é, com o mesmo sentido (x2) e com diferentes sentidos (x2). Em relação ao segundo cenário e visto estarmos perante uma disposição cruzada é impossível obter grupos mistos dado que apenas uma pista pode estar ativa simultaneamente. Ainda assim foram criados 4 grupos de pistas para aproximação e partida que representam as diferentes pistas lógicas. Por fim no terceiro e último cenário foram criados 12 grupos de apenas uma pista. Embora as várias pistas nunca se cruzem, as mesmas não possuem distâncias suficiente para serem utilizadas paralelamente.

Com a criação dos grupos criados é necessário ativar um grupo para que o mesmo possa ser utilizado pelas aeronaves. De forma a avaliar a escolha de grupos foram testados três cenários, um em que o rácio entre partidas e aterragem é 1:1, outro em que o cenário é 1:0 a favorecer as partidas e, por fim, um cenário com rácio de 0:1 a favorecer as chegadas. A tabela representada em

Resultados

6.2 mostra para cada tuplo Aeroporto-Cenário os valores estimados do quão ótimo será o melhor e o pior grupo de pistas.

Tabela 6.2: Resultado dos valores dados aos vários grupos de pistas para cada cenário.

	Cenário 1	Cenário 2	Cenário 3
KCMH	0.65	1	1
	0.55	0.1	0.1
KCAK	0.65	1	1
	0.65	0.1	0.1
KCLE	0.65	1	1
	0.65	0.1	0.1

Para o aeroporto *KCMH*, quando em simulação de possível equilíbrio entre aterragens e partidas (cenário 1), mostrou-se capaz de optar por um grupo constituído por dois tipos de aproximação diferente (10R para partida, 28R para chegada). Inversamente ao melhor grupo encontra-se o grupo constituído por apenas pistas de partidas apenas conseguiu 0.55. Para ambos os aeroportos *KCAK* e *KCLE*, dado que apenas possuem uma pista, a mesma terá de ser usadas para as duas operações.

Tabela 6.3: Tabela demonstrativa das pistas presentes nos melhores e piores grupos para cada tuplo Aeroporto-Cenário.

	Cenário 1	Cenário 2	Cenário 3
KCMH	10R partida	28R partida	10L chegada
	28R chegada	10R partida	28L chegada
	10L partida	10L chegada	10L partida
	10R partida	28L chegada	28L partida
KCAK	23 partida	23 partida	23 chegada
	10 chegada	10 chegada	10 partida
KCLE	28 partida	28 partida	28 chegada
	6L chegada	6R chegada	6L partida

É possível, através da Tabela 6.3, a visualização dos melhores e piores grupos de pistas existentes para cada tuplo Aeroporto-Cenário. Para os múltiplos aeroportos é possível notar que diferentes demandas resultam numa diferenciação de grupos de pistas a serem utilizados. Num primeiro aeroporto, *KCMH*, é possível observar que, para casos extremos de demanda (unicamente partidas ou unicamente chegadas), o aeroporto é capaz por optar por grupos constituídos por apenas pistas com a operação precisa. No entanto, para o mesmo aeroporto e para situações em que o aeroporto possui aeronaves a pretender aterrar e partir, o algoritmo utilizado é capaz de escolher grupos de pistas que contêm pistas para essas mesmas operações. Como seria de esperar, os grupos de pistas com pior utilidade mostram-se com operações opostas à pretendida. Para os aeroportos com grupos de apenas uma pista (*KCAK* e *KCLE*), o algoritmo embora seja capaz de optar pelas pistas e operações que melhor se adequam em casos extremos, apenas consegue utilizar uma única pista em situações de demanda equilibrada. A pista pertencente ao grupo ativado (23 partida e 28

partida no *KCAK* e *KCLE* respetivamente) terá obrigatoriamente de ser utilizada para ambas as operações existentes.

Embora os resultados se tenham mostrados satisfatórios, a forma de prever o grupo de pistas ainda precisa de uns ajustes nas variáveis de forma a diminuir a discrepância existente entre a plataforma e a realidade.

6.2 Gestão de Filas de Aterragem em Caso de Congestionamento

Sendo um dos objetivos primordiais da atual dissertação o melhoramento da gestão do espaço aéreo foi necessário avaliar os resultados que advêm das diferentes abordagens implementadas. De seguida é feita uma descrição dos vários cenários de teste assim como os vários resultados obtidos para cada um dos cenários.

6.2.1 Cenários de Testes

Ao longo do processo para melhorar a gestão do espaço aéreo do aeroporto foram desenvolvidas dois tipos de gestão de filas de espera. Os dois tipos de gestão assemelham-se entre si no facto que partilham o mecanismo da gestão das aeronaves ao longo dos diferentes níveis na fila, isto é, fazem cumprir as várias regras pela *FAA* e são *FIFO (First In First Out)*. A diferença entre as diferentes abordagens centra-se na forma como é dada permissão às aeronaves para aterrar. Se numa abordagem mais conservadora, apenas é concedida a permissão para aterrar à próxima aeronave caso a pista se encontre livre, na versão aprimorada é realizada uma tentativa de sequenciar das várias aeronaves de forma a que se consiga diminuir o tempo entre aterragem para uma determinada pista. De forma a obter a taxa de aterragem de um determinado aeroporto em situação de congestionamento, foram criadas oito aeronaves no espaço aéreo em que a todas as pistas se encontram fechadas. Foram utilizadas oito aeronaves para prevenir que as aeronaves que se encontrassem nas filas de esperas saíssem do espaço aéreo controlados pelo aeroportos. Após as várias aeronaves se estabilizarem nos respetivos níveis da fila de espera, as pistas eram liberadas novamente. O tempo de simulação, usado posteriormente, para cálculo da taxa de aterragem é definido após a primeira aeronave obter permissão para aterrar. É ainda relevante enunciar que as oito aeronaves são distribuídas igualmente pelas filas de espera ativas.

6.2.2 Resultados

Nesta sub-secção são apresentados os vários resultados obtidos para os vários testes realizados. Para cada cenário foram realizados cinco testes, sendo que a amostra mostrou-se admissível dada a magnitude do desvio padrão.

- Primeiro Cenário (*KCMH*)

Através das Tabelas 6.4 e 6.5 é possível notar um melhoramento bastante significativo quer em taxa de aterragem quer em tempo médio que uma aeronave que se encontra em último

Resultados

na fila de espera demora a ter permissão para iniciar a aproximação (ATRTLP - Average Time from Request To Land Permission).

Tabela 6.4: Resultados obtidos relativos às taxas de aterragem relativas ao aeroporto *KCMH*.

	Taxa de Aterragem	
	Média	Desvio Padrão
Versão Conservadora	13.395	0.3882
Versão Sequencial	44.12	1.926

Tabela 6.5: Tempo que as últimas aeronaves esperam entre pedir para aterrizar e obterem permissão no aeroporto de *KCMH*.

	ATRTLP	
	Média	Desvio Padrão
Versão Conservadora	00:26:49	00:00:22
Versão Sequencial	00:08:02	00:00:35

O aumento na ordem dos 314% é justificado por parte da abordagem sequencial é justificado pela velocidade da aeronave e pela distância das rotas de aproximação. Ao sequenciar as várias aeronaves é possível que se perca 8 ou 9:40 minutos, dependendo da rota de aproximação, apenas com uma aeronave. Em consequência a ter múltiplas aeronaves a aterrizar, o tempo de espera até ter permissão para aterrizar diminui numa escala semelhante. É de valor relembrar que, sendo as aeronaves criadas no local de espera (*fixpoint*), este tempo assemelha-se bastante com o tempo que a aeronave fica em espera na fila. Embora se tenha visto um melhoramento substancial na taxa de aterragem do aeroporto, este melhoramento poderá ser ainda melhor com uma melhor previsão de tempo que uma determinada aeronave demorará a aterrizar.

- Segundo Cenário (*KCLE*)

À semelhança do primeiro cenário, também o segundos resultados mostraram-se satisfatórios com um aumento de 326% para a abordagem sequencial (ver Tabela 6.6). A taxa de aterragem não se encontra nos mesmos valores que no primeiro cenário dado que apenas existe uma pista de aterragem, o que se pode confirmar com o aumento do tempo de espera para ter permissão para aterrizar (ver Tabela 6.7).

Tabela 6.6: Resultados obtidos relativos às taxas de aterragem relativas ao aeroporto *KCLE*.

	Taxa de Aterragem	
	Média	Desvio Padrão
Versão Conservadora	6.15	0.2179
Versão Sequencial	20.09	1.2024

Embora os resultados obtidos se mostrem razoáveis, uma taxa de vinte aeronaves mostra-se ainda aquém da realidade. No entanto é preciso notar que esta taxa é conseguida num caso

Resultados

Tabela 6.7: Tempo que as últimas aeronaves esperam entre pedir para aterrar e obterem permissão no aeroporto de *KCLE*.

	ATRTLP	
	Média	Desvio Padrão
Versão Conservadora	01:07:20	00:02:10
Versão Sequencial	00:20:12	00:01:25

de congestionamento onde as aeronaves não se encontram separadas, tornando mais difícil maximizar o fluxo dado os tempos perdidos entre descidas de níveis da fila.

- Terceiro Cenário (*KCAK*)

Os resultados obtidos para o aeroporto *KCAK* encontram-se apresentados nas Tabelas 6.8 e 6.9. Com um aumento substancial na taxa de aterragem, o aeroporto é capaz de reduzir o tempo de espera de 69 minutos para uns 18 minutos.

Tabela 6.8: Resultados obtidos relativos às taxas de aterragem relativas ao aeroporto *KCAK*.

	Taxa de Aterragem	
	Média	Desvio Padrão
Versão Conservadora	6.02	0.0657
Versão Sequencial	21.7628	0.8183

Tabela 6.9: Tempo que as últimas aeronaves esperam entre pedir para aterrar e obterem permissão no aeroporto de *KCAK*.

	ATRTLP	
	Média	Desvio Padrão
Versão Conservadora	01:09:14	00:01:08
Versão Sequencial	00:18:35	00:00:53

Com a realização dos testes acima descritos tornou-se claro que a utilização da abordagem capaz de sequenciar as aeronaves representa uma melhoria na taxa de aterragem. No entanto, embora a abordagem a usar contribua imensamente para a melhoria da taxa de aterragem, o comprimento da rota de aproximação (Segundo vs Terceiro Cenário), a configuração das pistas (Primeiro Vs Segundo Cenário) assim como a velocidade das aeronaves são capazes de alterar significativamente a mesma.

6.3 Gestão do Espaço Aéreo do Aeroporto com Auxílio do *TRACON*

Ao contrário dos testes descritos na secção 6.2, onde se partiu de um congestionamento do espaço aéreo do aeroporto, os testes da presente secção pretendem *alimentar* o aeroporto com aeronaves espaçadas entre si tal como acontece na realidade. Nesta secção será descrito o cenário da simulação assim como os resultados obtidos para os vários testes realizados.

6.3.1 Cenários de Teste

Para ser possível a realização destes testes é necessário a utilização de dois agentes *ATC*, sendo que um irá assumir o papel de controlador "*Tracon*" e o segundo o papel de controlador de aeroporto. Para a realização deste teste foi escolhido o aeroporto *KCMH* dado a sua boa *performance* anterior na secção 6.2 e o número de *STARs* que se possuem rota de ligação. De modo a preparar o início da simulação são lançadas oito aeronaves distribuídas igualmente pelos *STARs* existentes (*BREMN* e *GAILL*).

A simulação tem início aquando da estabilização das diferentes aeronaves nos seus respetivos níveis. Após o início da simulação as aeronaves vão, tendo em conta as regras especificadas pela *FAA*, saindo das filas de espera e aproximando-se do espaço aéreo do aeroporto. Ao entrar no espaço aéreo do aeroporto, as mesmas pedem permissão para aterrar sendo que o pedido pode ser negado ou aceite. Após a entrada no espaço aéreo do aeroporto, o controlador *TRACON* deixa de ter qualquer controlo sobre a aeronave. A simulação termina quando a última aeronave das oito iniciais atravessar o aterrar no aeroporto de origem.

De forma a se tornar possível a comparação entre os resultados obtidos neste cenário com os analisados na secção 6.2, foram utilizadas as mesmas aeronaves, *Beech Baron 58*, para a execução dos testes deste cenário.

6.3.2 Resultados

Tabela 6.10: Resultados obtidos no aeroporto de destino com auxilio do *TRACON*.

	Taxa de Aterragem	
	Média	Desvio Padrão
Versão C/ <i>TRACON</i>	26.604	3.559
Versão S/ <i>TRACON</i>	44.122	1.926

Após uma análise dos resultados obtidos para o presente cenário, foi constatada uma diminuição considerável da taxa de aterragem face aos resultados obtidos numa versão de congestionamento a nível do aeroporto. A diferença resultante dos vários testes realizados podem ser explicados pelas várias incertezas, que se encontram presentes na realização do presente teste, que incutem um taxa de erro considerável para um melhor aproveitamento da sequencialização das aeronaves.

De forma a conseguir assegurar a separação em todos os instantes da viagem entre o local utilizado para a realização de manobras de espera até ao aeroporto, o controlador aéreo *TRACON* calcula com base na velocidade de cruzeiro das aeronaves, o intervalo de tempo necessário entre libertação das aeronaves. No entanto, após a entrada no espaço aéreo do aeroporto, a velocidade das aeronaves é reduzida substancialmente. Tendo conhecimento da desaceleração e da distância a ser percorrida ser inferior à calculada, o controlador adiciona uma margem de erro de um minuto ao intervalo de tempo entre libertação de aeronaves. Tendo os vários *STARs* um total de quatro aeronaves, após ser dada permissão à última aeronave, foram *perdidos* quatro minutos.

Resultados

Comparando com o facto de, para o mesmo aeroporto e em condições semelhantes (aterragens sequenciais), as aeronaves conseguirem aterrar com intervalos de aproximadamente dois minutos seria possível adicionar mais duas aeronaves às oito.

Resultados

Capítulo 7

Conclusões e Trabalho Futuro

No presente capítulo são expostas as várias conclusões obtidas assim como possíveis trabalhos a serem realizados num futuro.

7.1 Conclusões

Ao longo do século XX e do presente século, o mundo da aviação tem vindo a assistir a uma maior procura e uso por parte da sociedade. A comodidade de poder fazer voos longos em relativamente pouco tempo aliada à segurança oferecida pelos meios de transporte e a diminuição dos custos fazem deste meio de transporte um dos mais apetecíveis. No entanto, os aeroportos não estão a conseguir acompanhar o ritmo de crescimento deste meio de transporte, o que acaba por dificultar a realização das tarefas dos controladores aéreos assim como comprometer a integridade do espaço aéreo (acidentes de aviação ou criação de um congestionamento). Com base nos problemas acima descritos torna-se cada vez mais urgente a construção de plataformas capazes de auxiliar os controladores ou de tomar controlo de uma região de espaço aéreo de forma a que se consiga manter um espaço aéreo seguro.

Como forma de solucionar o problema na plataforma foram implementados várias funcionalidades capazes de melhorar a gestão do espaço aéreo em zonas de aproximação finais a aeroportos (*STARs*) e no espaço aéreo dos aeroportos.

Para a eventualidade de acontecer um congestionamento do espaço aéreo, foi criado um sistema de gestão de filas de espera de forma a que se consiga manter as aeronaves organizadas por tempo indefinido. Com o intuito de conseguir mitigar o congestionamento aéreo no período de tempo mais curto possível, foi criada a funcionalidade de sequenciar aeronaves para aterrar. Esta funcionalidade aliada à gestão de filas de espera demonstrou bons resultados visto que ao contrário de em [Camara, 2013] já é levado em conta a componente temporal das aeronaves.

Com o objetivo de prevenir o congestionamento do espaço aéreo de um aeroporto, e à semelhança do que acontece na realidade, as aeronaves podem esperar pela sua vez nos *STARs*. Com

isso em mente, foi implementado uma gestão do espaço aéreo ao nível dos *STARs* capaz de prever quando é que uma determinada aeronave pode ou não sair em direção ao aeroporto. De modo a conseguir obter essa gestão e previsão foi necessária a implementação de um novo controlador. Esta conjunto de funcionalidades mostrou-se eficaz mas ficou ainda aquém do que acontece na realidade. A razão para se encontrar desfasado da realidade explica-se pelo facto de o controlador responsável pela gestão a nível dos *STARs* não conseguir controlar a aeronave no espaço aéreo do aeroporto.

De forma a aproximar a plataforma da realidade atual, foi necessário implementar o uso de rotas de aproximação reais. Aproveitou-se ainda esta rotas reais para ser criado um algoritmo capaz de criar grupos de rotas que podem ser usadas de forma simultânea.

7.2 Trabalho Futuro

Ao longo da realização da presente dissertação foi sendo adotado um novo objetivo - diminuição da discrepância entre a realidade e a plataforma. Com a introdução deste objetivo não principal foi necessário a alteração do plano de tarefas a realizar que tinha sido previamente desenhado. Assim como possíveis trabalhos a serem realizados no âmbito da gestão do espaço aéreo, tem-se a criação de um módulo capaz de criar/alterar rotas várias aeronaves a circular no espaço aéreo de forma a minimizar/evitar o impacto causado pelos congestionamentos que possam aparecer. Seria também interessante o desenvolvimento de um protocolo de comunicação entre aeroportos cujo objetivo passaria pela troca de informação de modo a atrasar/cancelar ou mesmo adiantar voos. Não relacionado com o congestionamento aéreo, seria interessante a realização de um funcionalidade capaz de permitir a comutação entre as regra impostas pela *FAA* ou recomendadas pela *ICAO*.

Referências

- [Aero, 2016] Aero, M. (2016). Air traffic management. Disponível online em <https://max.aero/flight-training/private-pilot-training-program/learning-center/air-traffic-management/> (Acedido em Junho 2017).
- [Airbus, 2017] Airbus (2017). A statistical analysis of commercial aviation accidents 1958-2016. Disponível online em <http://www.skybrary.aero/bookshelf/books/4009.pdf> (Acedido em Julho 2017).
- [Airport-technology, 2017] Airport-technology (2017). Navcanatm - integrated air traffic management and control systems. Disponível online em <http://www.airport-technology.com/contractors/traffic/navcanatm/> (Acedido em Junho 2017).
- [Andreatta and Romanin-Jacur, 1987] Andreatta, G. and Romanin-Jacur, G. (1987). Aircraft flow management under congestion. *Transportation Science*, 21(4):249–253.
- [Balakrishnan and Chandran, 2010] Balakrishnan, H. and Chandran, B. G. (2010). Algorithms for scheduling runway operations under constrained position shifting. *Oper. Res.*, 58(6):1650–1665.
- [Bayen et al., 2004] Bayen, A. M., Tomlin, C. J., Ye, Y., and Zhang, J. (2004). An approximation algorithm for scheduling aircraft with holding time. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, volume 3, pages 2760–2767 Vol.3.
- [Camara, 2013] Camara, Á. (2013). Controlo de tráfego aéreo usando o microsoft flight simulator x. Master’s thesis, Department of Informatics Engineering, University of Coimbra, Coimbra, Portugal.
- [Câmara et al., 2014] Câmara, Á., Silva, D. C., Abreu, P. H., and Oliveira, E. (2014). Comparing a centralized and decentralized multi-agent approaches to air traffic control. In *Proceedings of the 28th European Simulation and Modelling Conference (ESM’2014), October 22-24, 2014, Porto, Portugal*, pages 189–193.
- [CANADA, 2017] CANADA, N. (2017). Overview of navcanatm. Disponível online em <http://www.navcanada.ca/EN/media/Publications/NAVCANatm-Backgrounder-EN.pdf> (Acedido em Junho 2017).
- [Chaimatanan et al., 2015] Chaimatanan, S., Delahaye, D., and Mongeau, M. (2015). Aircraft 4d trajectories planning under uncertainties. In *2015 IEEE Symposium Series on Computational Intelligence*, pages 51–58.
- [Dearden, 2015] Dearden, L. (2015). The one chart that shows what causes fatal plane crashes. Disponível online em

REFERÊNCIAS

- <http://www.independent.co.uk/travel/news-and-advice/the-one-chart-that-shows-what-causes-fatal-plane-crashes-10494952.html> (Acedido em Junho 2017).
- [Djeumou Fomeni et al., 2017] Djeumou Fomeni, F., Lulli, G., and Zografos, K. (2017). An optimization model for assigning 4d-trajectories to flights under the tbo concept. In *Twelfth USA/Europe ATM R&D Seminar*, pages 1–10.
- [ENAIRE, 2017] ENAIRE (2017). Phases of a flight. Disponível online em <http://www.enaire.es/csee/Satellite/SeguridadOperacionalNA/en/Page/1237551632185/1228215409288/> (Acedido em Junho 2017).
- [Eurocontrol, 2016] Eurocontrol (2016). Eurocontrol seven-year forecast. Disponível online em <http://www.eurocontrol.int/sites/default/files/content/documents/official-documents/forecasts/seven-year-flights-service-units-forecast-2016-2022-september-2016.pdf> (Acedido em Junho 2017).
- [FAA, 2000] FAA (2000). Flight plan requirements for helicopter operational under instrument flight rules. Disponível online em <https://www.gpo.gov/fdsys/pkg/FR-2000-01-21/pdf/00-1326.pdf> (Acedido em Junho 2017).
- [FAA, 2006] FAA (2006). Fact sheet – co-located tracons (terminal radar approach control). Disponível online em https://www.faa.gov/news/fact_sheets/news_story.cfm?contentkey=4009 (Acedido em Junho 2017).
- [FAA, 2012] FAA (2012). Taxiway nomenclature convention. Disponível online em https://www.faa.gov/airports/engineering/engineering_briefs/media/eb-89.pdf (Acedido em Junho 2017).
- [FAA, 2015a] FAA (2015a). Aeronautical information manual official guide to basic flight information and atc procedures. Acedido em Novembro de 2016.
- [FAA, 2015b] FAA (2015b). En route automation modernization (eram). Disponível online em https://www.faa.gov/air_traffic/technology/eram/ (Acedido em Junho 2017).
- [FAA, 2016a] FAA (2016a). Air traffic by the numbers. Disponível online em https://www.faa.gov/air_traffic/by_the_numbers/ (Acedido em Junho 2017).
- [FAA, 2016b] FAA (2016b). Fact sheet – standard terminal automation replacement system. Disponível online em https://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=21034/ (Acedido em Junho 2017).
- [FAA, 2017] FAA (2017). Aeronautical information services - national flight data center (nfdc). Disponível online em <https://nfdc.faa.gov/nfdcApps/services/airspaceFixes/fixes.jsp> (Acedido em Junho 2017).
- [Flightgear, 2014] Flightgear (2014). Atc phraseology. Disponível online em http://wiki.flightgear.org/ATC_phraseology (Acedido em Junho 2017).
- [Freeman, 2016] Freeman, P. (2016). Abandoned & little-known airfields. Disponível online em http://www.members.tripod.com/airfields_freeman/TN/Airfields_TN_E.htm (Acedido em Junho 2017).

REFERÊNCIAS

- [Grabbe et al., 2009] Grabbe, S., Sridhar, B., and Mukherjee, A. (2009). Sequential traffic flow optimization with tactical flight control heuristics. *Journal of Guidance, Control, and Dynamics*, 32(3):810–820.
- [Hart et al., 1972] Hart, P. E., Nilsson, N. J., and Raphael, B. (1972). A formal basis for the heuristic determination of minimum cost paths. *SIGART Bull.*, (37):28–29.
- [Houston, 2016a] Houston, S. (2016a). Nextgen in a nutshell. Disponível online em <https://www.thebalance.com/nextgen-in-a-nutshell-282561> (Acedido em Junho 2017).
- [Houston, 2016b] Houston, S. (2016b). What are the different types of notams in aviation? Disponível online em <https://www.thebalance.com/what-is-a-notam-282687> (Acedido em Junho 2017).
- [ICAO, 2010] ICAO (2010). Operation of aircraft. annex 6.
- [ICAO, 2013] ICAO (2013). *Global Operational Data Link Document*. International Civil Aviation Organization, second edition.
- [Isidore, 2015] Isidore, C. (2015). What’s the safest way to travel. Disponível online em <http://money.cnn.com/2015/05/13/news/economy/train-plane-car-deaths/> (Acedido em Junho 2017).
- [IVAO, 2010] IVAO (2010). Faa/icao the differences in phraseology. Disponível online em http://peter-ftp.co.uk/aviation/misc-euroga/2016-faa_{_}icao.pdf (Acedido em Junho 2017).
- [IVAO, 2016] IVAO (2016). Holding stack management. Disponível online em https://www.iviao.aero/training/documentation/books/APC_Stack_manage.pdf (Acedido em Junho 2017).
- [Kiss-Tóth and Takács, 2014] Kiss-Tóth, C. and Takács, G. (2014). A dynamic programming approach for 4d flight route optimization. In *2014 IEEE International Conference on Big Data (Big Data)*, pages 24–28.
- [Martyna, 2013] Martyna, J. (2013). Runway scheduling with holding pattern and service priority. *Automatics/Automatyka*, 16(2):137.
- [Metacraft, 2006] Metacraft (2006). Virtual radar client. Disponível online em <http://www1.metacraft.com/VRC/> (Acedido em Junho 2017).
- [Metacraft, 2017] Metacraft (2017). veram. Disponível online em <http://www1.metacraft.com/VRC/> (Acedido em Junho 2017).
- [Microsoft, 2008] Microsoft (2008). Esp sdk overview. Disponível online em <https://msdn.microsoft.com/en-us/library/cc526948.aspx> (Acedido em Junho 2017).
- [Mogford et al., 1995] Mogford, R., Guttman, J. A., Morrow, S. L., and Kopardekar, P. (1995). The complexity construct in air traffic control: A review and synthesis of the literature. Technical report, CTA INC MCKEE CITY NJ.
- [Morris, 2017] Morris, H. (2017). How many planes are there in the world right now? Disponível online em <http://www.telegraph.co.uk/travel/travel-truths/how-many-planes-are-there-in-the-world/> (Acedido em Junho 2017).

REFERÊNCIAS

- [NashvilleCFI, 2011] NashvilleCFI (2011). Holds. Disponível online em <https://nashvillecfi.com/instrument/holds.html> (Acedido em Junho 2017).
- [NAV Portugal, 2017] NAV Portugal, E. (2017). Movimentos ifr totais na riv de lisboa. Disponível online em <https://www.nav.pt/nav/quem-somos/dados-de-tr%C3%A1fego> (Acedido em Junho 2017).
- [NextGen, 2007] NextGen (2007). Nextgen concept of operations, version 2.0. Disponível online em https://info.aiaa.org/tac/AASG/ACOTC/Shared%20Documents/NextGen_v2.0.pdf (Acedido em Junho 2017).
- [Ng et al., 2009] Ng, H. K., Grabbe, S., and Mukherjee, A. (2009). *Design and Evaluation of a Dynamic Programming Flight Routing Algorithm Using the Convective Weather Avoidance Model*, chapter Design and Evaluation of a Dynamic Programming Flight Routing Algorithm Using the Convective Weather Avoidance Model. Guidance, Navigation, and Control and Co-located Conferences. American Institute of Aeronautics and Astronautics. 0.
- [Nolan, 2010] Nolan, M. S. (2010). *FUNDAMENTALS OF AIR TRAFFIC CONTROL*. Delmar Cengage Learning, fifth edition.
- [Pfeil and Balakrishnan, 2012] Pfeil, D. M. and Balakrishnan, H. (2012). Identification of robust terminal-area routes in convective weather. *Transportation Science*, 46(1):56–73.
- [Raytheon, 2017] Raytheon (2017). Standard terminal automation replacement system (stars) - faa. Disponível online em http://www.raytheon.com/capabilities/products/stars_faa/ (Acedido em Junho 2017).
- [Rodrigues et al., 2015] Rodrigues, C., Silva, D. C., Rossetti, R. J. F., and Oliveira, E. (2015). Distributed flight simulation environment using flight simulator x. In *Proceedings of the 10th Iberian Conference on Information Systems and Technologies, June 17-20 2015, Águeda, Portugal*, pages 1293–1297.
- [Salehipour et al., 2013] Salehipour, A., Modarres, M., and Naeni, L. M. (2013). An efficient hybrid meta-heuristic for aircraft landing problem. *Computers & Operations Research*, 40(1):207 – 213.
- [Santos, 2010] Santos, A. (2010). Autonomous Intelligent Vehicle Adaptation and Performance Analysis in Flight Simulator X. Master’s thesis, Faculty of Engineering, University of Porto, Porto, Portugal.
- [School, 2016] School, L. F. (2016). Hold patterns and hold pattern entry procedures. Disponível online em <http://www.langleflyingschool.com/Pages/Holds%20and%20Hold%20Entries1.html#Standard%20Holding%20Pattern> (Acedido em Junho 2017).
- [SESAR, 2016] SESAR (2016). Why sesar? Disponível online em <http://www.sesarju.eu/discover-sesar/why-sesar> (Acedido em Junho 2017).
- [Shaw, 2014] Shaw, D. R. J. (2014). History of flight. Disponível online em <https://www.grc.nasa.gov/www/k-12/UEET/StudentSite/historyofflight.html> (Acedido em Junho 2017).

REFERÊNCIAS

- [Silva, 2011] Silva, D. C. (2011). *Cooperative Multi-Robot Missions: Development of a Platform and a Specification Language*. PhD thesis, Faculty of Engineering, University of Porto, Porto, Portugal.
- [Silva et al., 2014] Silva, D. C., Abreu, P. H., Reis, L. P., and Oliveira, E. (2014). Development of a Flexible Language for Mission Description for Multi-Robot Missions. *Information Sciences*, 288:27–44.
- [Silva et al., 2017] Silva, D. C., Abreu, P. H., Reis, L. P., and Oliveira, E. (2017). Development of Flexible Languages for Scenario and Team Description in Multi-Robot Missions. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 31(1):69–86. DOI: 10.1017/S0890060416000184.
- [Silva, 2008] Silva, R. (2008). Communication Layer in an Aviation Simulator. Master’s thesis, Faculty of Engineering, University of Porto, Porto, Portugal.
- [Skybrary, 2010] Skybrary (2010). Flight phase taxonomy. Disponível online em http://www.skybrary.aero/index.php/Flight_Phase_Taxonomy (Acedido em Junho 2017).
- [Skybrary, 2015] Skybrary (2015). Controller pilot data link communications (cpdlc). Disponível online em [http://www.skybrary.aero/index.php/Controller_Pilot_Data_Link_Communications_\(CPDLC\)](http://www.skybrary.aero/index.php/Controller_Pilot_Data_Link_Communications_(CPDLC)) (Acedido em Junho 2017).
- [Skybrary, 2016] Skybrary (2016). Notice to airman. Disponível online em http://www.skybrary.aero/index.php/Notice_To_Airmen (Acedido em Junho 2017).
- [Soomer and Franx, 2008] Soomer, M. and Franx, G. (2008). Scheduling aircraft landings using airlines’ preferences. *European Journal of Operational Research*, 190(1):277 – 291.
- [Sousa, 2010] Sousa, P. D. (2010). Autonomous Air Traffic Control for Intelligent Vehicles using Microsoft Flight Simulator X. Master’s thesis, Faculty of Engineering, University of Porto, Porto, Portugal.
- [Stentz, 1995] Stentz, A. (1995). The focussed d* algorithm for real-time replanning. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2, IJCAI’95*, pages 1652–1659, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- [Szczerba et al., 2000] Szczerba, R. J., Galkowski, P., Glicktein, I. S., and Ternullo, N. (2000). Robust algorithm for real-time route planning. *IEEE Transactions on Aerospace and Electronic Systems*, 36(3):869–878.
- [U.S. Department of Transportation Federal Aviation Administration, 2010] U.S. Department of Transportation Federal Aviation Administration (2010). Departure procedures and separation. Disponível online em <http://tfmlearning.faa.gov/Publications/atpubs/ATC/atc0309.html> (Acedido em Junho 2017).
- [U.S Senate Joint Economic Committee, 2008] U.S Senate Joint Economic Committee (2008). Your flight has been delayed again - flight delays cost passengers, airlines, and the u.s. economy billions. Disponível online em <https://www.jec.senate.gov/public/index.cfm/democrats/flightdelayscostpassengers> (Acedido Junho 2017).
- [Vanhoenacker, 2015] Vanhoenacker, M. (2015). In flight. Disponível online em <https://www.nytimes.com/interactive/2015/05/14/opinion/14-in-flight-mark-vanhoenacker.html> (Acedido em Junho 2017).

REFERÊNCIAS

- [VATSIM.net, 2017] VATSIM.net (2017). About vatsim. Disponível online em <https://www.vatsim.net/about> (Acedido em Junho 2017).
- [Vecchiola et al., 2008] Vecchiola, C., Grosso, A., and Boccalatte, A. (2008). Agentservice; a framework to develop distributed multiagent systems. *Int. J. Agent-Oriented Softw. Eng.*, 2(3):290–323.
- [Vranas, 1994] Vranas, Peter B.; Bertsimas, D. J. O. A. R. (1994). The multi-airport ground-holding problem in air traffic control. *Oper. Res.*, 42(2):249–261.
- [Wald et al., 2015] Wald, A., Fay, C., and Gleich, R. (2015). *Introduction to Aviation Management*. LIT Verlag, third edition.
- [Wang et al., 2010] Wang, L., Zhang, Z., and Liu, J. (2010). Multiple airport ground holding dynamic model and optimal algorithm basing flight priority. In *IEEE ICCA 2010*, pages 1081–1085.
- [Zhang and Zhang, 2006] Zhang, A. and Zhang, Y. (2006). Airport capacity and congestion when carriers have market power. *Journal of Urban Economics*, 60(2):229 – 247.
- [Zillies, 2013] Zillies, J. L. (2013). Multiobjective 4d optimization of a trajectory-based air traffic management. In *2013 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, pages 1–11.

Anexo A

Anexos

Encontra-se em anexo as várias modificações realizadas nos ficheiros XSD e nas respetivas classes.

A.1 Alterações nos Ficheiros de Linguagem

Apresentam-se aqui as alterações aos dialetos SDL e MDL, apresentando os elementos adicionados na especificação dos dialetos (usando XSD).

A.1.1 Ficheiros relativos à *SDL*

```
1  <xs:element name="approachRoutes">
2    <xs:complexType>
3      <xs:sequence>
4        <xs:element ref="approachToRunway" minOccurs="0" maxOccurs="unbounded"/>
5      </xs:sequence>
6    </xs:complexType>
7  </xs:element>
8  <xs:element name="approachToRunway">
9    <xs:complexType>
10     <xs:sequence>
11       <xs:element ref="approach" minOccurs="0" maxOccurs="unbounded"/>
12     </xs:sequence>
13     <xs:attribute type="xs:string" name="runway" use="required"/>
14   </xs:complexType>
15 </xs:element>
16 <xs:element name="approach">
17   <xs:complexType>
18     <xs:sequence>
19       <xs:element ref="approachWaypoints" minOccurs="0" />
20       <xs:element ref="missedApproach" minOccurs="0" />
21     </xs:sequence>
22   </xs:complexType>
```

Anexos

```
23 </xs:element>
24 <xs:element name="approachWaypoints">
25   <xs:complexType>
26     <xs:sequence>
27       <xs:element ref="approachWaypoint" minOccurs="0" maxOccurs="unbounded" />
28     </xs:sequence>
29   </xs:complexType>
30 </xs:element>
31 <xs:element name="missedApproach">
32   <xs:complexType>
33     <xs:sequence>
34       <xs:element ref="approachWaypoint" minOccurs="0" maxOccurs="unbounded"/>
35     </xs:sequence>
36   </xs:complexType>
37 </xs:element>
38 <xs:element name="approachWaypoint">
39   <xs:complexType>
40     <xs:sequence>
41       <xs:element ref="coordinates"/>
42     </xs:sequence>
43     <xs:attribute type="xs:string" name="name" use="required"/>
44     <xs:attribute type="xs:boolean" name="hasHolding" use="required"/>
45     <xs:attribute name="holdingDegrees" type="xs:string" />
46     <xs:attribute name="holdingDir" type="xs:string" />
47   </xs:complexType>
48 </xs:element>
49 <xs:element name="airspace">
50   <xs:complexType>
51     <xs:sequence>
52       <xs:element ref="waypoints"/>
53       <xs:element ref="airways"/>
54     </xs:sequence>
55   </xs:complexType>
56 </xs:element>
57
58 <xs:element name="airways">
59   <xs:complexType>
60     <xs:sequence>
61       <xs:element ref="airway" maxOccurs="unbounded"/>
62     </xs:sequence>
63   </xs:complexType>
64 </xs:element>
65
66 <xs:element name="waypoints">
67   <xs:complexType>
68     <xs:sequence>
69       <xs:element ref="waypoint" maxOccurs="unbounded"/>
70     </xs:sequence>
71   </xs:complexType>
```

```

72 </xs:element>
73
74 <xs:element name="airway">
75   <xs:complexType>
76     <xs:sequence>
77       <xs:element ref="origin" />
78       <xs:element ref="destination"/>
79       <xs:element ref="trafficrodensity" />
80     </xs:sequence>
81     <xs:attributeGroup ref="idGroup"/>
82   </xs:complexType>
83 </xs:element>
84
85 <xs:element name="waypoint">
86   <xs:complexType>
87     <xs:sequence>
88       <xs:element ref="latitude" />
89       <xs:element ref="longitude"/>
90     </xs:sequence>
91     <xs:attributeGroup ref="idGroup"/>
92     <xs:attribute name="name" use="required"/>
93   </xs:complexType>
94 </xs:element>
95
96 <xs:element name="stars">
97   <xs:complexType>
98     <xs:sequence>
99       <xs:element ref="star" minOccurs="0" maxOccurs="unbounded" />
100     </xs:sequence>
101   </xs:complexType>
102 </xs:element>
103 <xs:element name="star">
104   <xs:complexType>
105     <xs:sequence>
106       <xs:element ref="coordinates" minOccurs="0" maxOccurs="unbounded" />
107       <xs:element ref="connections" minOccurs="0" maxOccurs="unbounded" />
108     </xs:sequence>
109     <xs:attribute name="name" type="xs:string" />
110     <xs:attribute name="holdingDegrees" type="xs:string" />
111     <xs:attribute name="holdingDir" type="xs:string" />
112   </xs:complexType>
113 </xs:element>
114 <xs:element name="connections">
115   <xs:complexType>
116     <xs:sequence>
117       <xs:element ref="courses" minOccurs="0" maxOccurs="unbounded" />
118     </xs:sequence>
119   </xs:complexType>
120 </xs:element>

```

Anexos

```
121 <xs:element name="courses">
122   <xs:complexType>
123     <xs:sequence>
124       <xs:element ref="course" minOccurs="0" maxOccurs="unbounded" />
125     </xs:sequence>
126     <xs:attribute name="runway" form="unqualified" type="xs:string" />
127   </xs:complexType>
128 </xs:element>
129 <xs:element name="course">
130   <xs:complexType>
131     <xs:sequence>
132       <xs:element ref="Fixpoint" minOccurs="0" maxOccurs="unbounded" />
133     </xs:sequence>
134     <xs:attribute name="cid" type="xs:string" />
135   </xs:complexType>
136 </xs:element>
137 <xs:element name="Fixpoint">
138   <xs:complexType>
139     <xs:sequence>
140       <xs:element ref="coordinates" minOccurs="0" maxOccurs="unbounded" />
141     </xs:sequence>
142     <xs:attribute name="name" type="xs:string" />
143     <xs:attribute name="holdingDegrees" form="unqualified" type="xs:string" />
144     <xs:attribute name="holdingDir" form="unqualified" type="xs:string" />
145   </xs:complexType>
146 </xs:element>
147
148 <xs:element name="departures">
149   <xs:complexType>
150     <xs:sequence>
151       <xs:element ref="departure" minOccurs="0" maxOccurs="unbounded" />
152     </xs:sequence>
153   </xs:complexType>
154 </xs:element>
155
156 <xs:element name="departure">
157   <xs:complexType>
158     <xs:sequence>
159       <xs:element ref="fromRunway" minOccurs="0" maxOccurs="1" />
160       <xs:element ref="departureSequence" minOccurs="0" maxOccurs="1" />
161     </xs:sequence>
162   </xs:complexType>
163 </xs:element>
164
165 <xs:element name="fromRunway">
166   <xs:complexType>
167     <xs:sequence>
168       <xs:element ref="lrway" minOccurs="0" maxOccurs="unbounded"/>
169     </xs:sequence>
```

```

170     </xs:complexType>
171 </xs:element>
172
173 <xs:element name="lrway">
174     <xs:complexType>
175         <xs:attribute name="idr" form="unqualified" type="xs:string" />
176     </xs:complexType>
177 </xs:element>
178
179 <xs:element name="departureSequence">
180     <xs:complexType>
181         <xs:sequence>
182             <xs:element ref="sequenceAlternative" minOccurs="0" maxOccurs="unbounded" /
183                 >
184             </xs:sequence>
185         </xs:complexType>
186 </xs:element>
187 <xs:element name="sequenceAlternative">
188     <xs:complexType>
189         <xs:sequence>
190             <xs:element ref="Fixpoint" minOccurs="0" maxOccurs="unbounded" />
191             <xs:element ref="altitude" minOccurs="0" maxOccurs="unbounded" />
192         </xs:sequence>
193     </xs:complexType>
</xs:element>

```

Listagem A.1: Alterações no ficheiro scenario.xsd

A.1.2 Ficheiros relativos à MDL

```

1
2 <xs:element name="action">
3     <xs:complexType>
4         <xs:choice>
5             <xs:element ref="route"/>
6             <xs:element ref="searchPattern"/>
7             <xs:element ref="searchAreaDivisionMethod"/>
8             <xs:element ref="changeFormation"/>
9             <xs:element ref="vehicleType"/>
10        </xs:choice>
11    </xs:complexType>
12 </xs:element>
13 <xs:element name="route">
14     <xs:complexType>
15         <xs:sequence>
16             <xs:element ref="wpoint" minOccurs="0" maxOccurs="unbounded"/>
17         </xs:sequence>

```

Anexos

```
18     </xs:complexType>
19 </xs:element>
20 <xs:element name="wpoint">
21     <xs:complexType>
22         <xs:sequence>
23             <xs:element ref="altitude"/>
24             <xs:element ref="after" minOccurs="0"/>
25         </xs:sequence>
26         <xs:attribute name="name" type="xs:string" use="required"/>
27     </xs:complexType>
28 </xs:element>
```

Listagem A.2: Alterações no ficheiro mission.xsd